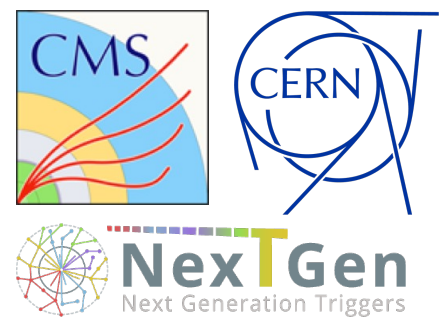


# Advancing the CMS Level-1 Trigger: Jet Tagging with DeepSets at the HL-LHC

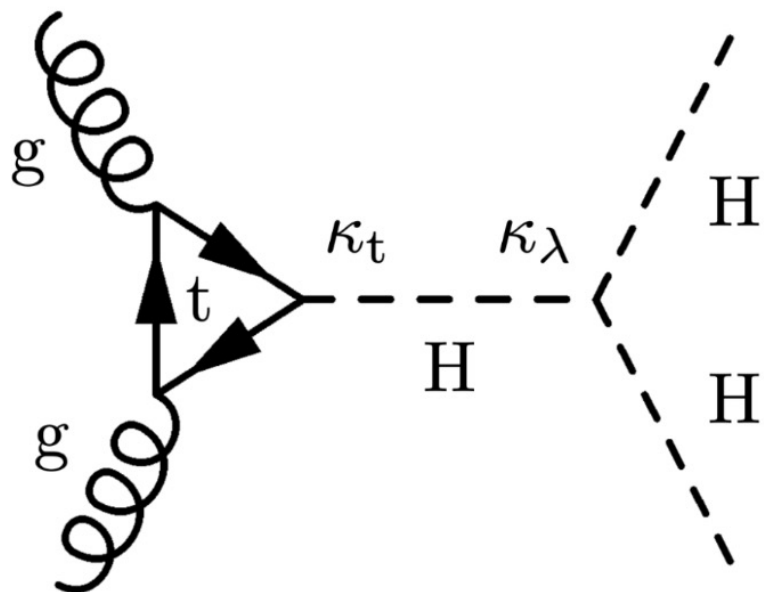
Christopher Brown  
On behalf of the CMS Collaboration



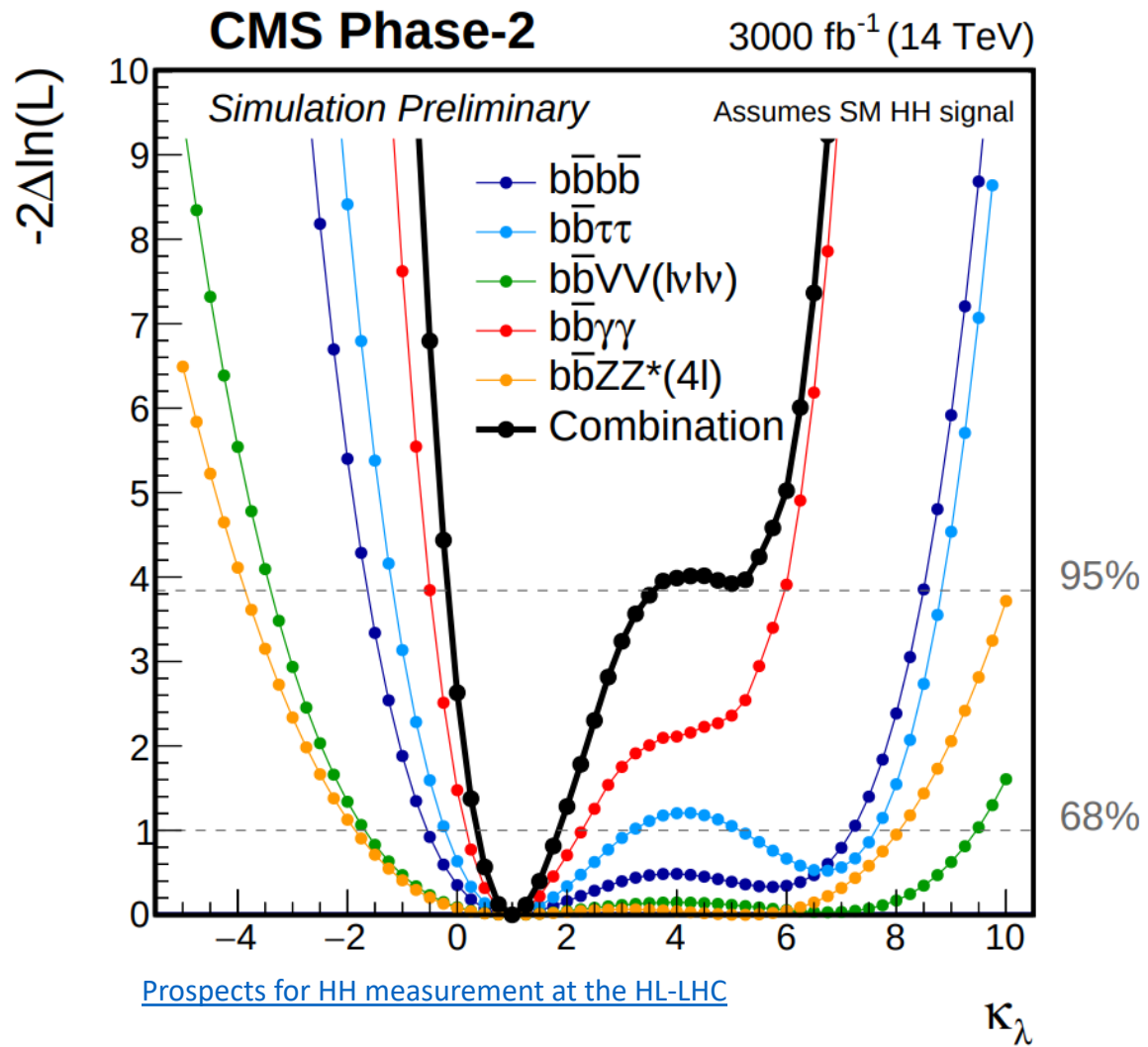
# Advancing the CMS Level-1 Trigger: **Jet Tagging** with DeepSets at the HL-LHC

Christopher Brown  
On behalf of the CMS Collaboration

Start at the end in the offline analysis

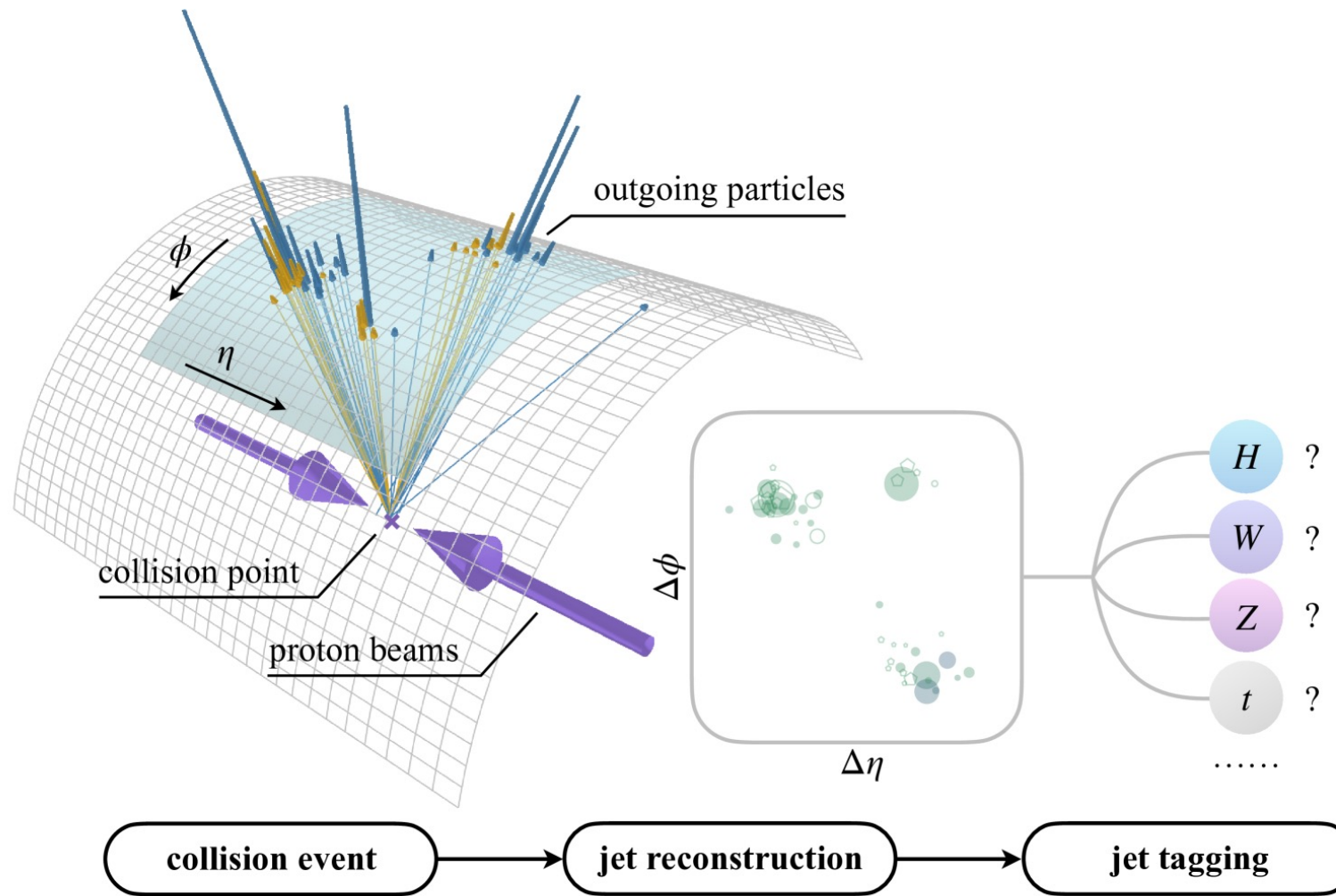


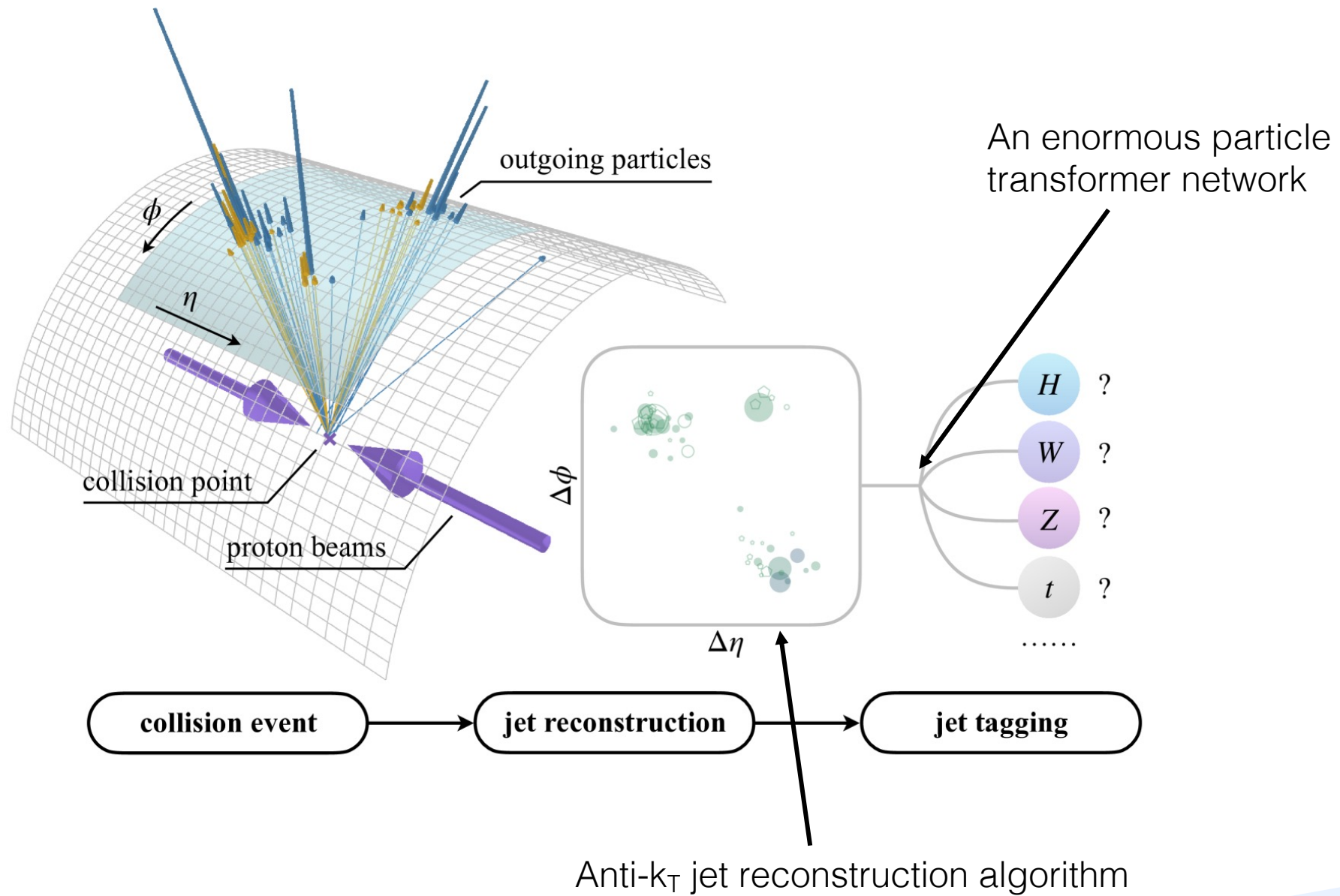
Want to measure  $\kappa_\lambda$ , the Higgs self coupling

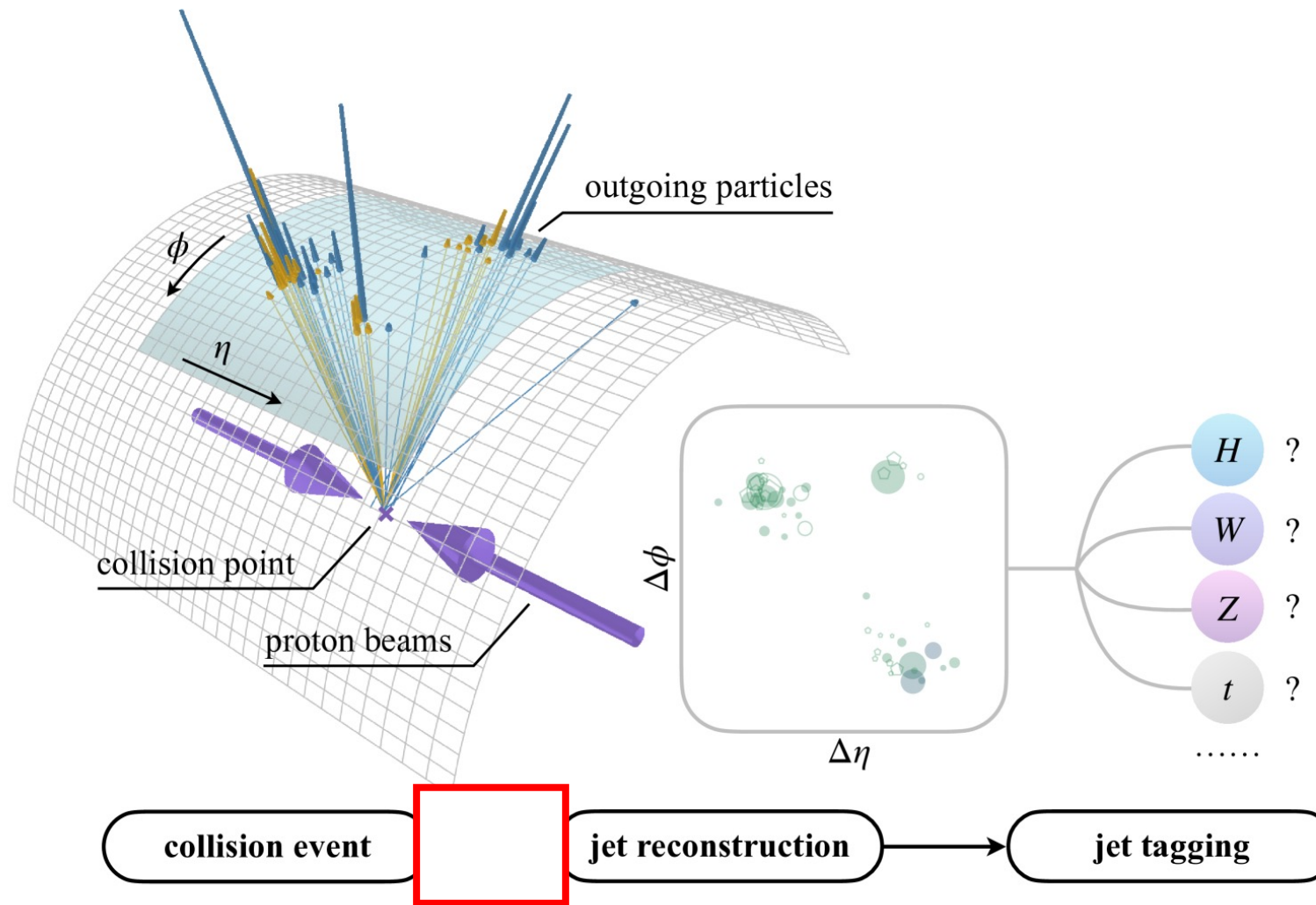


Need the events with specific decay channels

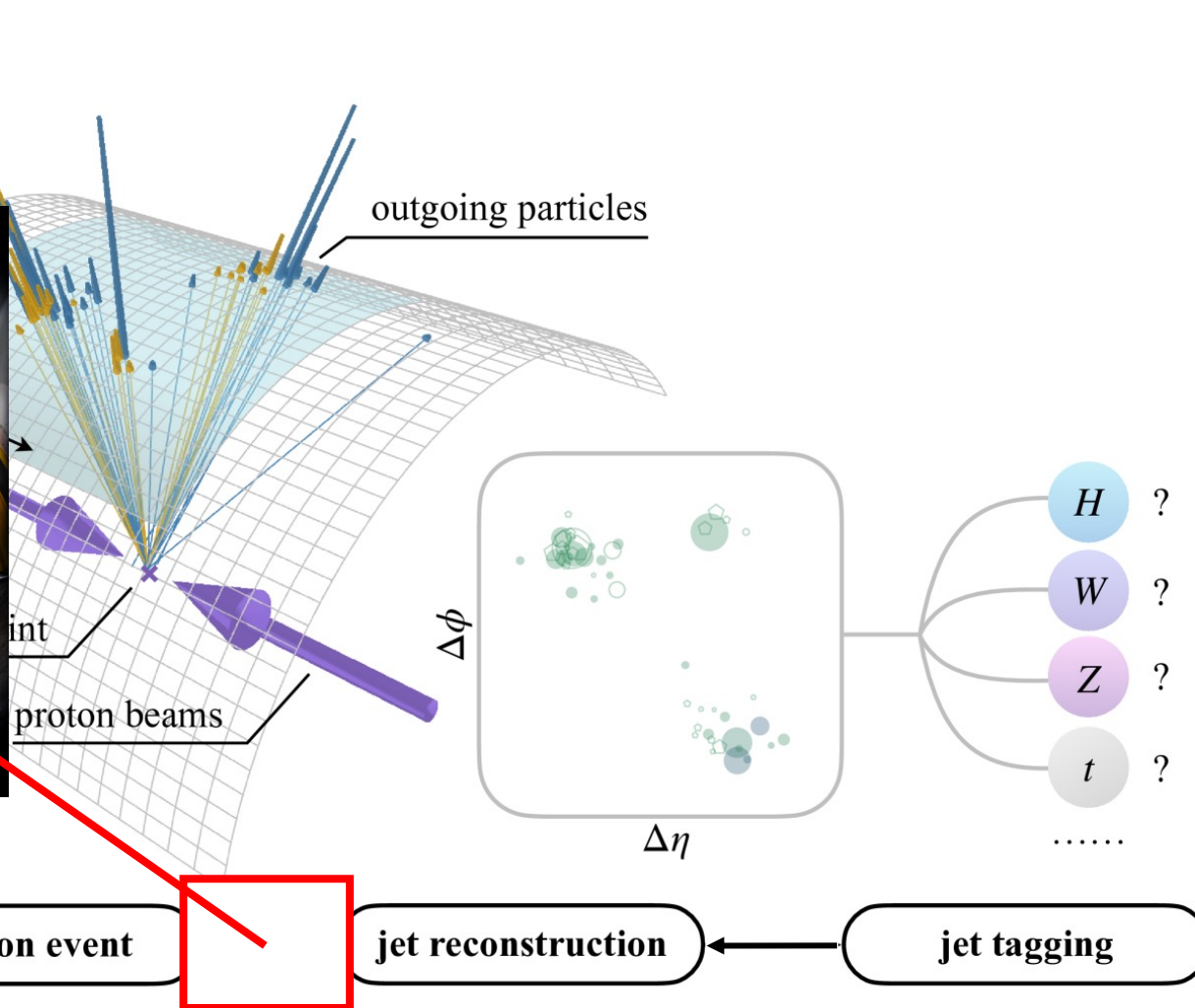
Need to know when the decay is b and when it's a  $\tau$



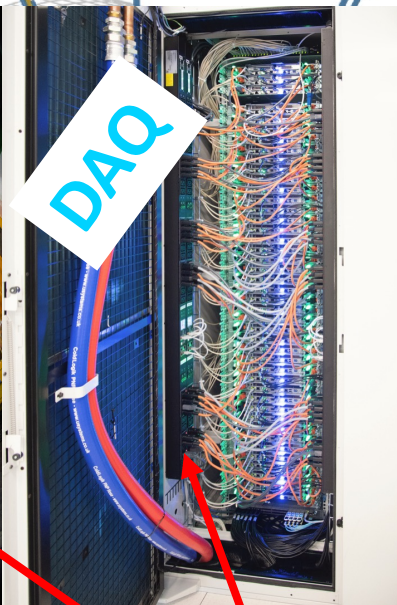




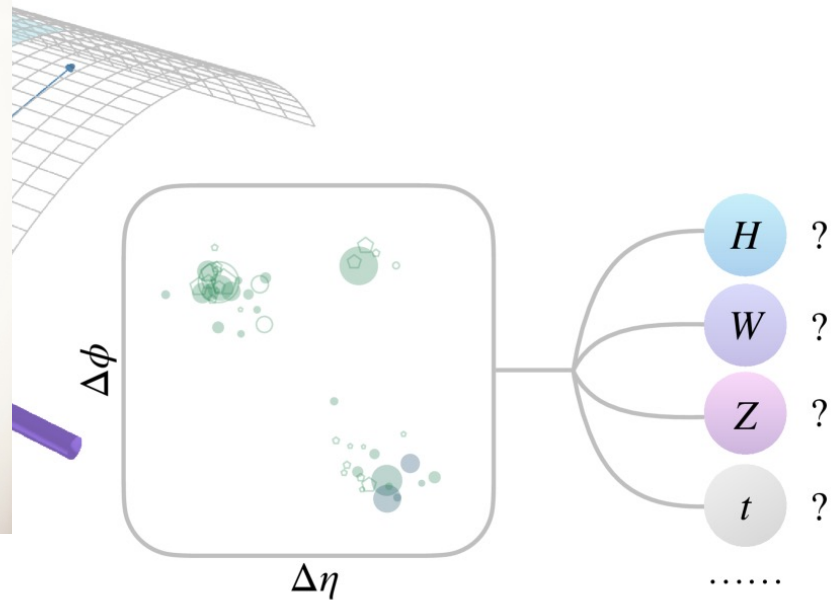








outgoing particles



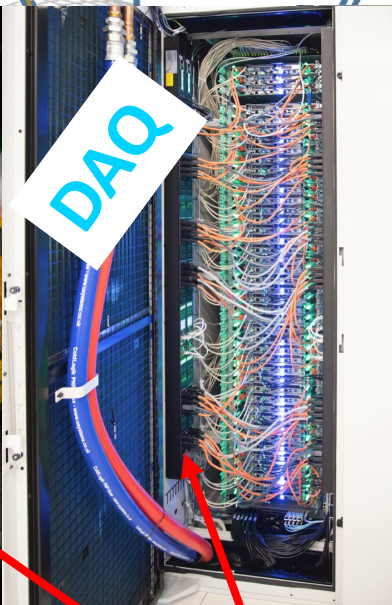
collision event

jet reconstruction

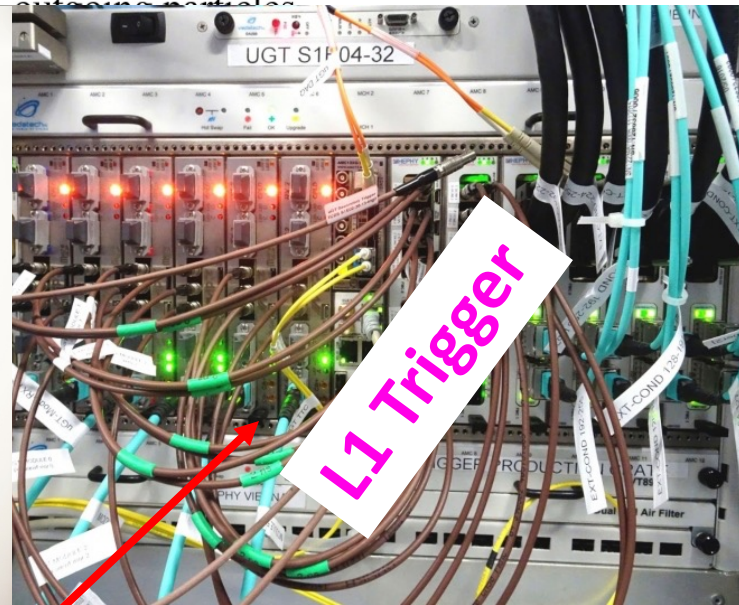
jet tagging



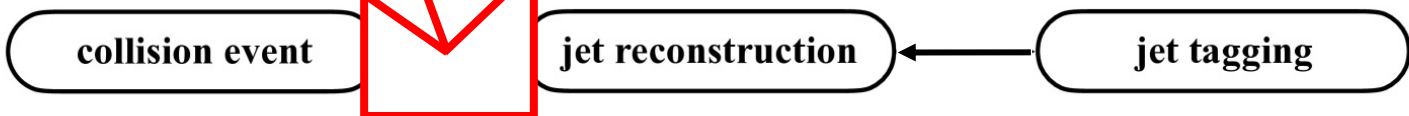
DETECTOR



DAQ



L1 Trigger



collision event

jet reconstruction

jet tagging

$\Delta\eta$

.....

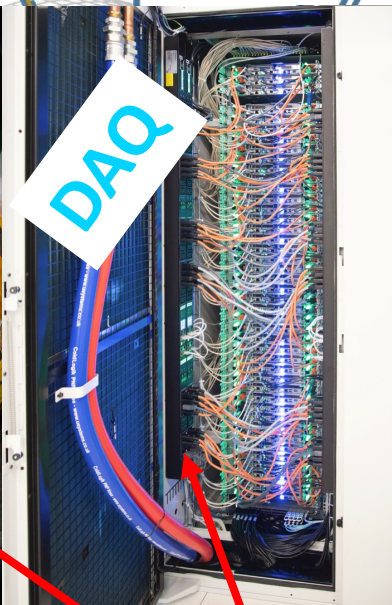
H ?  
W ?  
Z ?  
t ?



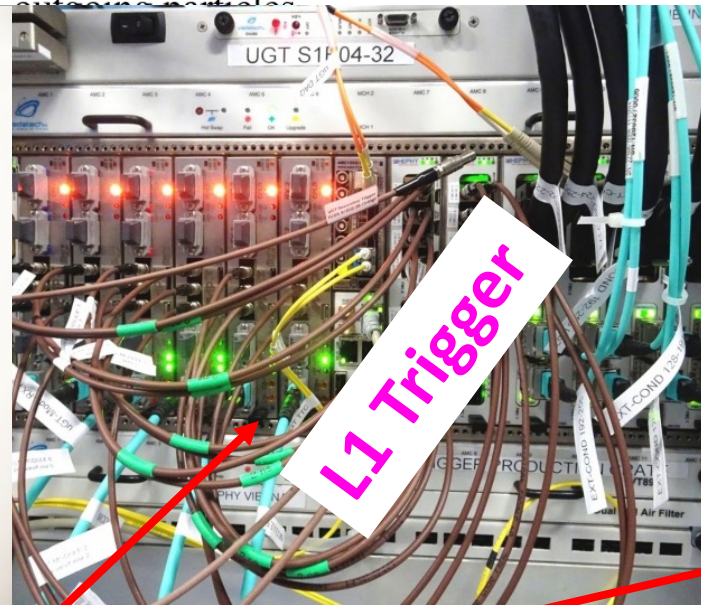




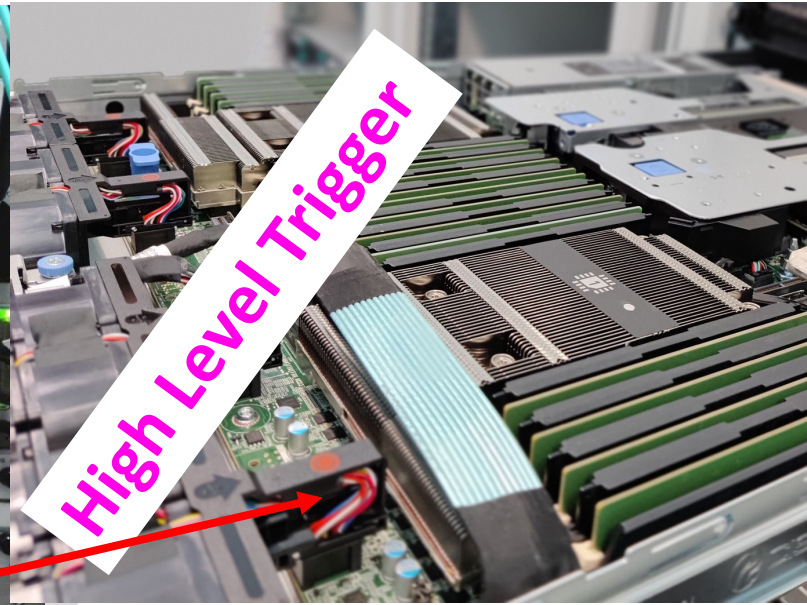
DETECTOR



DAQ



L1 Trigger

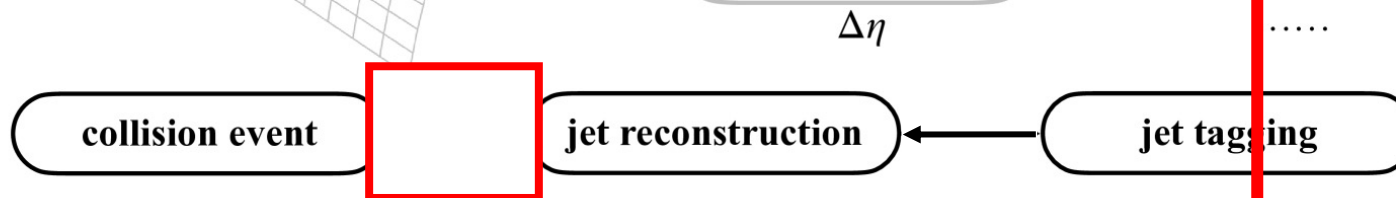
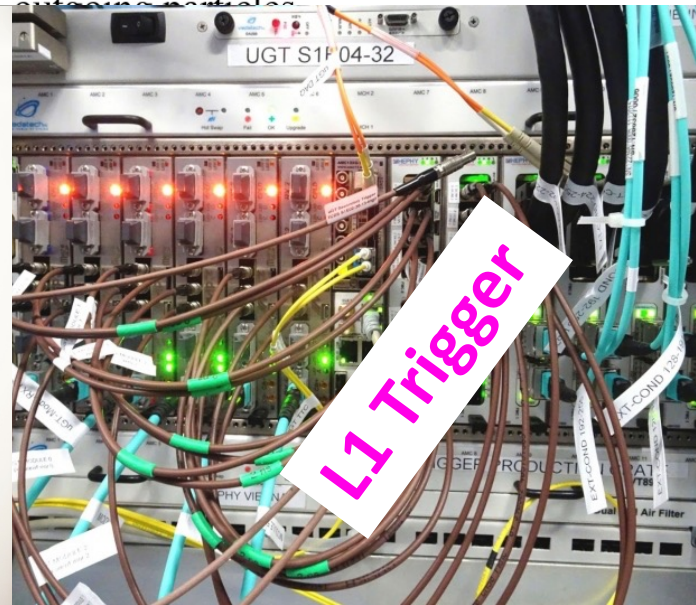


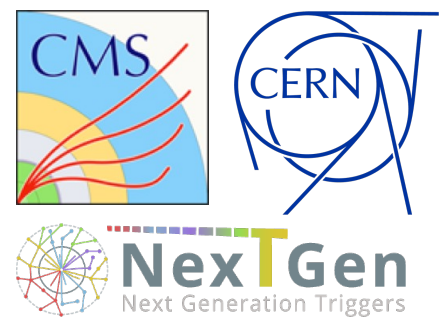
High Level Trigger





99% of the data stops here





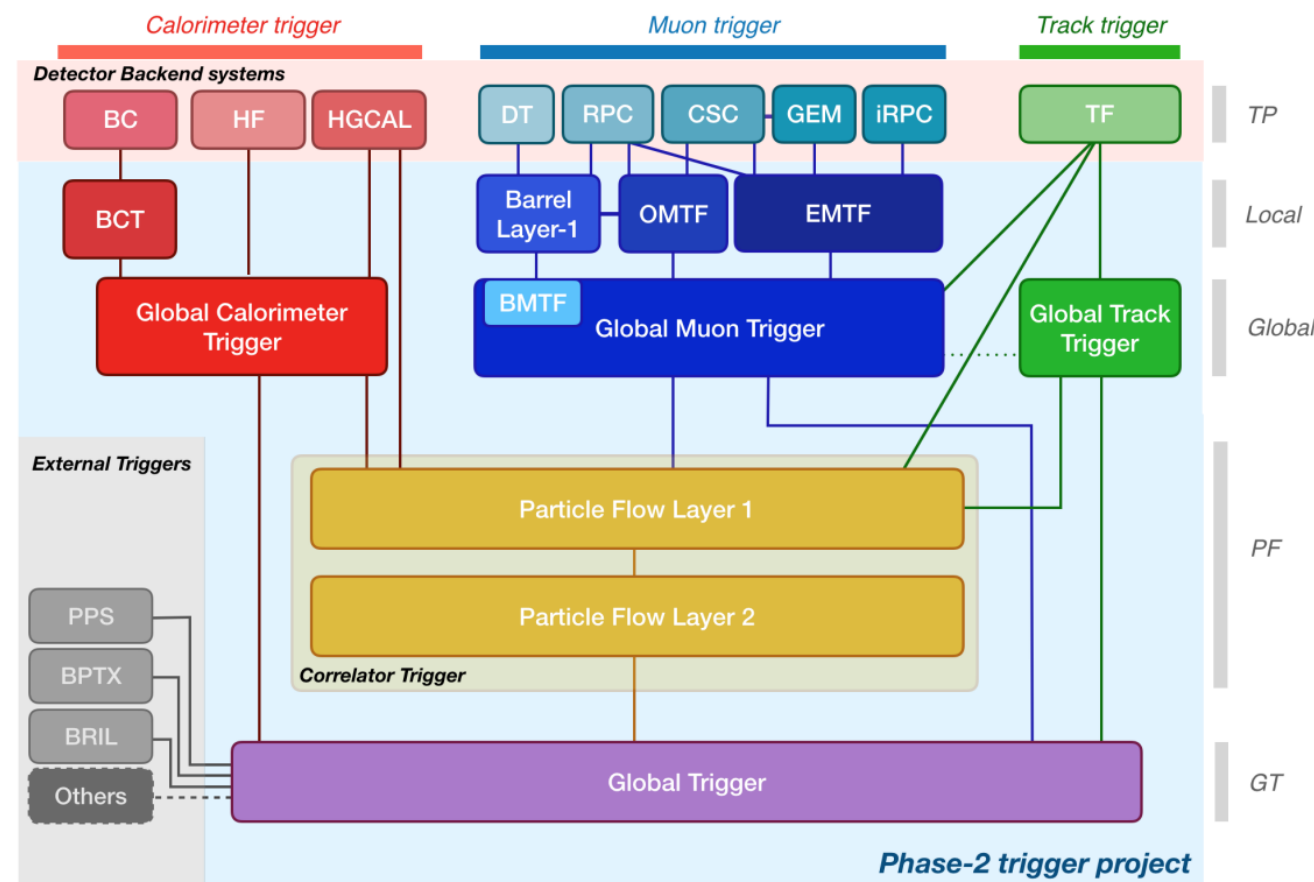
# Advancing the CMS Level-1 Trigger: Jet Tagging with DeepSets at the HL-LHC

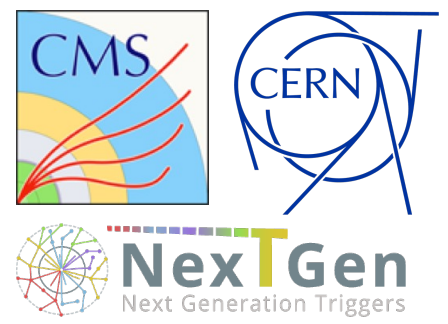
Christopher Brown  
On behalf of the CMS Collaboration

CMS Phase-2 Trigger gives us lots of tools

- Tracking  $p_T > 2 \text{ GeV}$
- High Granularity Calorimeter
- Particle Flow
- Pile Up per Particle Identification (PUPPI)
- Machine Learning

How do we use this information to tag jets?



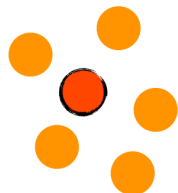


# Advancing the CMS Level-1 Trigger: Jet Tagging with DeepSets at the HL-LHC

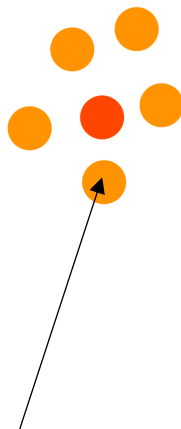
Christopher Brown  
On behalf of the CMS Collaboration



# Step 1 – Make some jets



Find the highest  $p_T$  candidate and set as seed

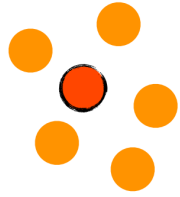


Particle Flow candidates matched to the primary vertex using the PUPPI algorithm

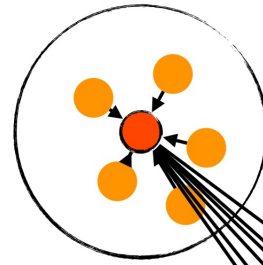
[The Particle Flow Algorithm in the Phase II upgrade of the CMS Level-1 Trigger](#)



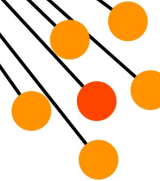
# Step 1 – Make some jets



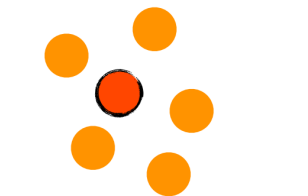
Find the highest  $p_T$  candidate and set as seed



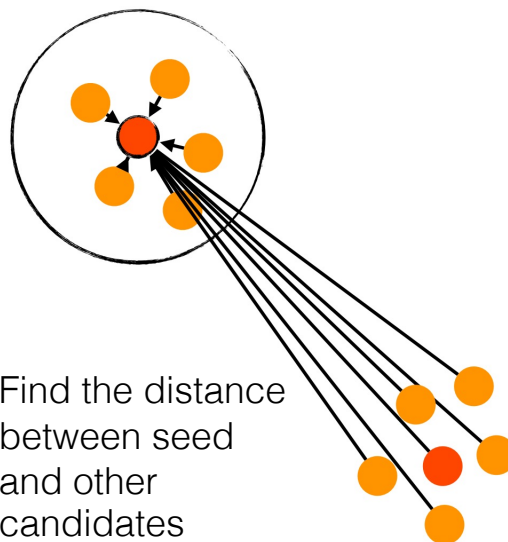
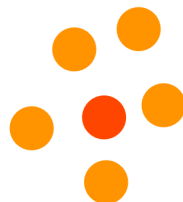
Find the distance between seed and other candidates



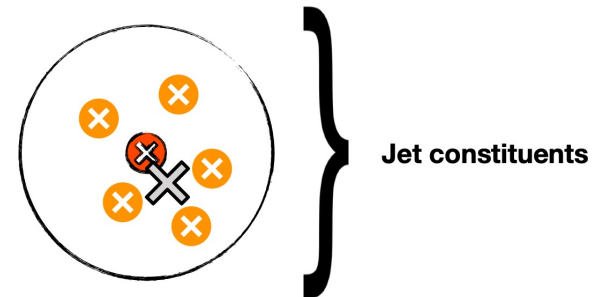
# Step 1 – Make some jets



Find the highest  $p_T$  candidate and set as seed



Find the distance between seed and other candidates

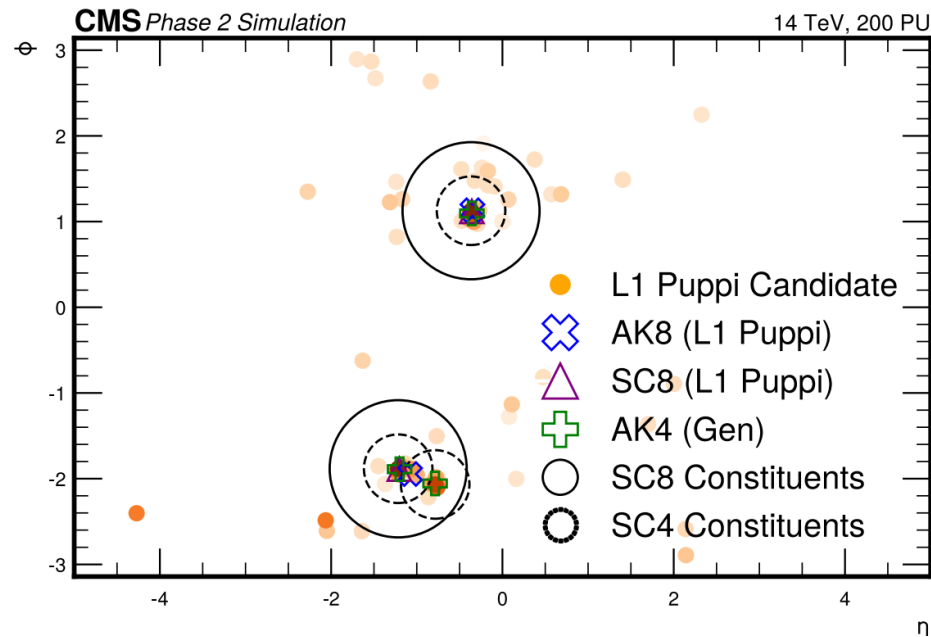
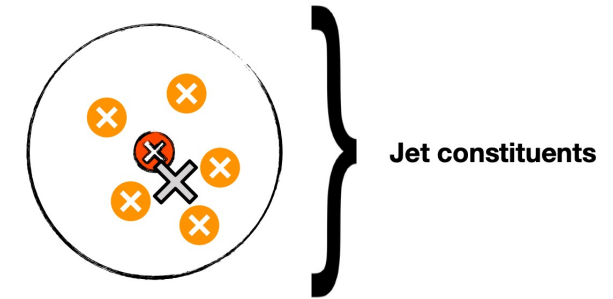
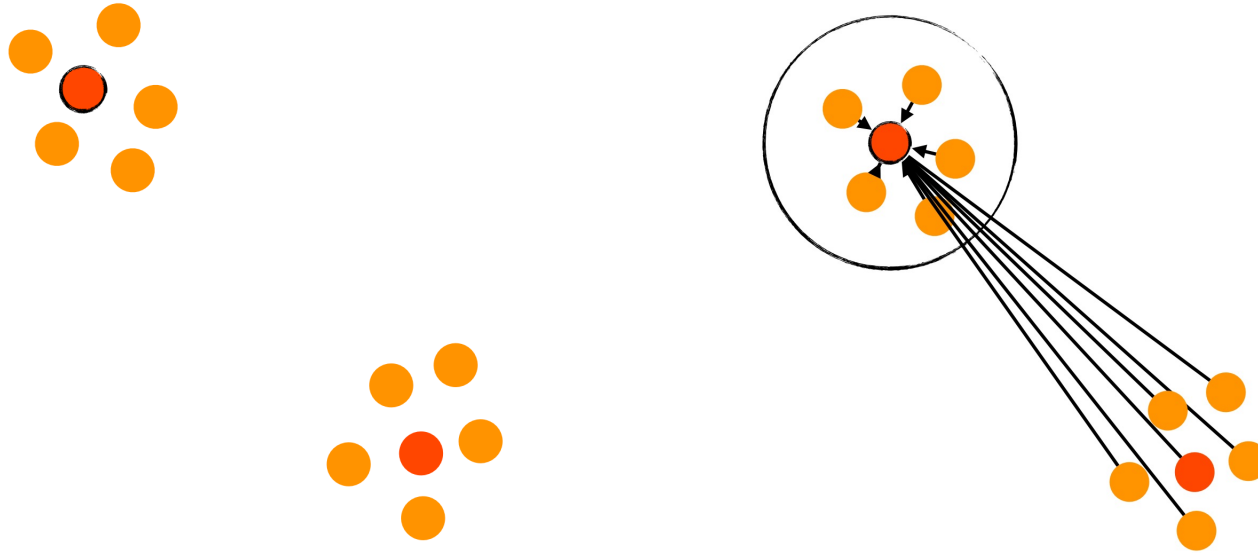


Use a radius to constrain the candidates and calculate jet axis and corrected  $p_T$

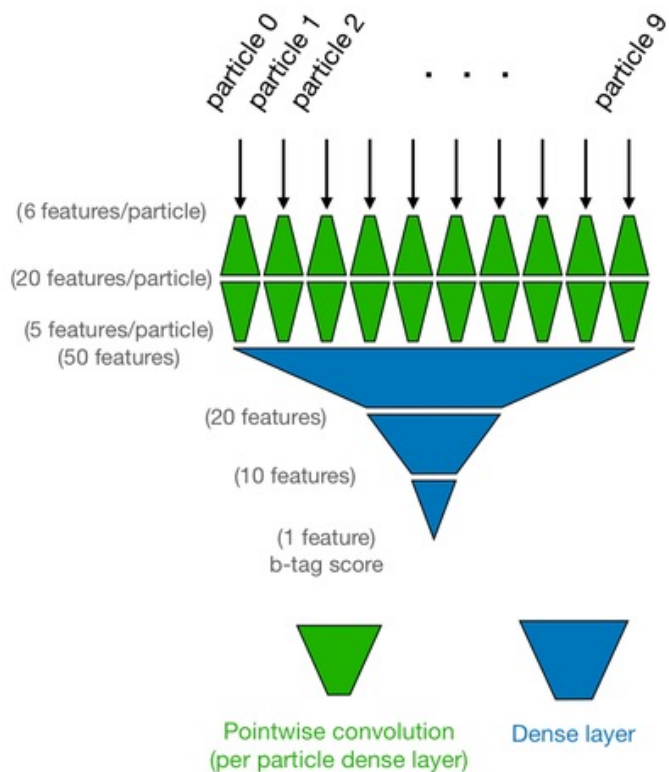


Mask found jet and repeat

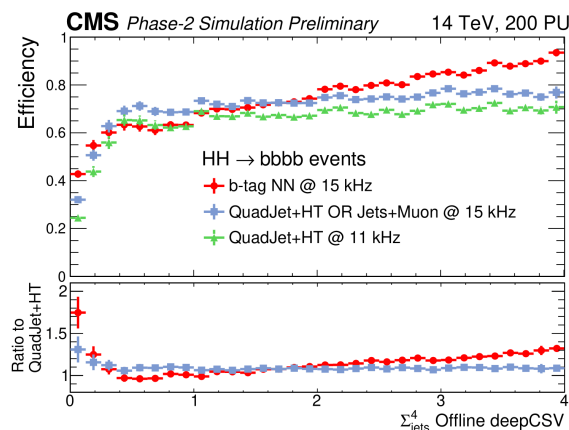
# Step 1 – Make some jets



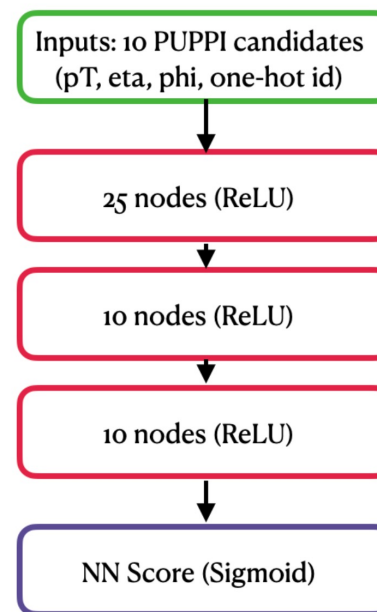
## Step 2 – Use the jet constituents to predict what made the jet



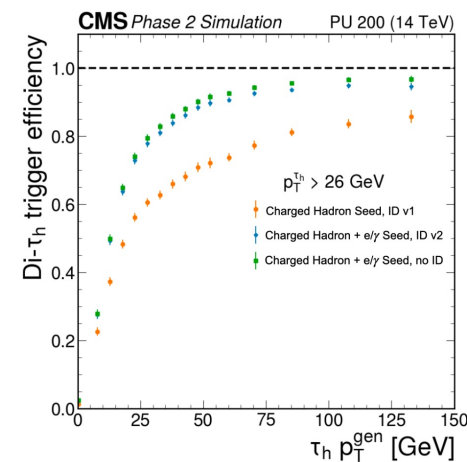
b jets



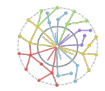
[Neural network-based algorithm for the identification of bottom quarks in the CMS Phase-2 Level-1 trigger](#)



hadronic tau jets



[NNPuppiTaus: PUPPI tau reconstruction in the Level-1 trigger with real-time machine learning](#)



## Step 2 – Use the jet constituents to predict what made the jet

Offline they don't have separate networks per class

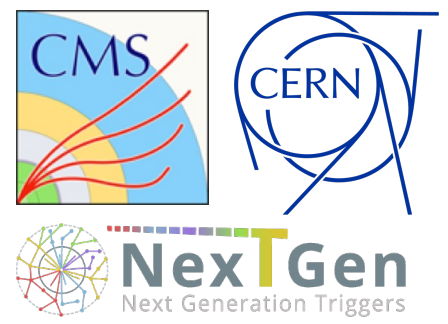
One network learns the **representation of a jet**

One network classifies it into **multiple different decay modes**

One network has **potentially fewer FPGA resources**

One network can do **additional tasks** for every jet regardless of their class



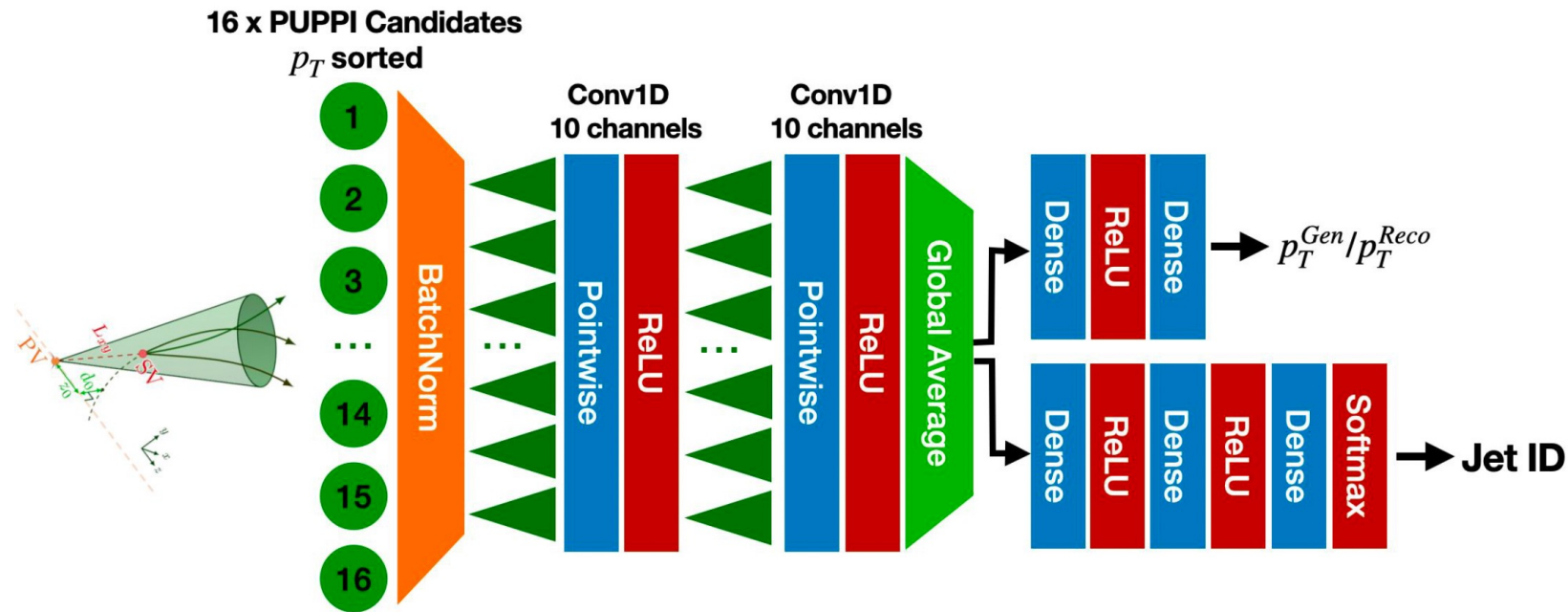


# Advancing the CMS Level-1 Trigger: Jet Tagging with DeepSets at the HL-LHC

Christopher Brown  
On behalf of the CMS Collaboration



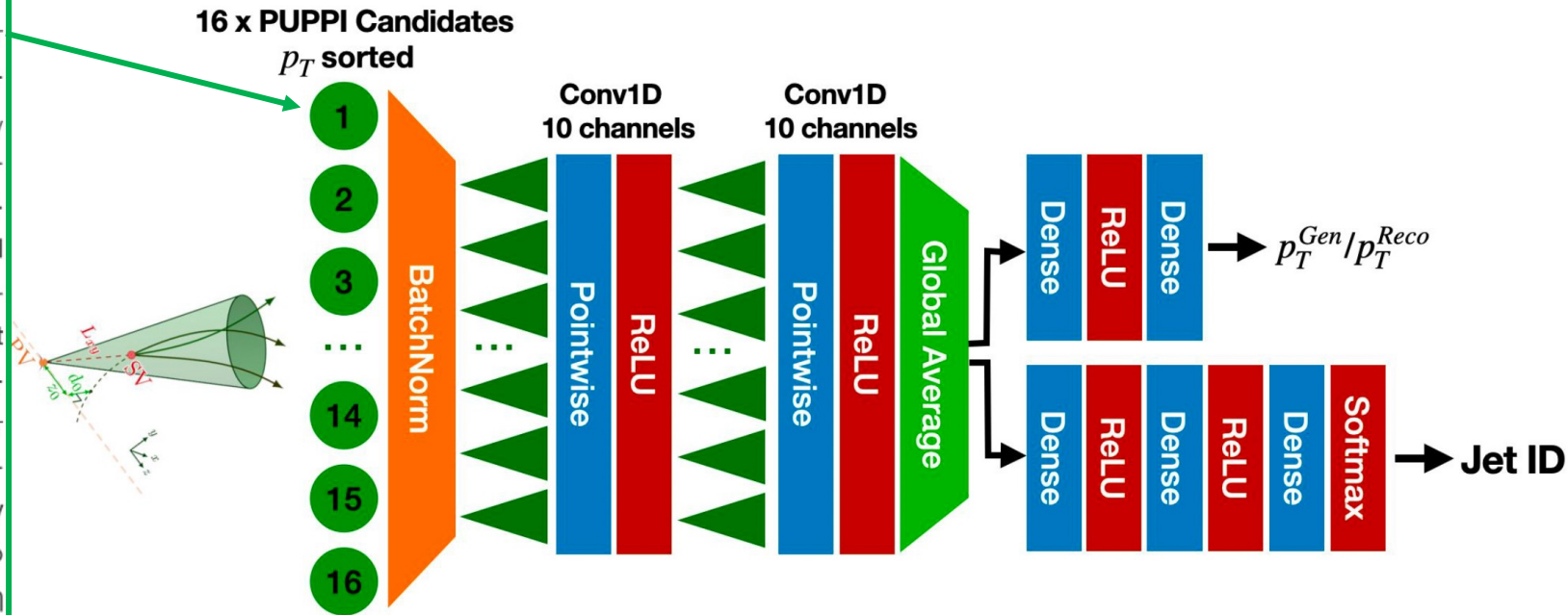
# One network



# One network

20 input features per candidate, up to 16 candidates per jet

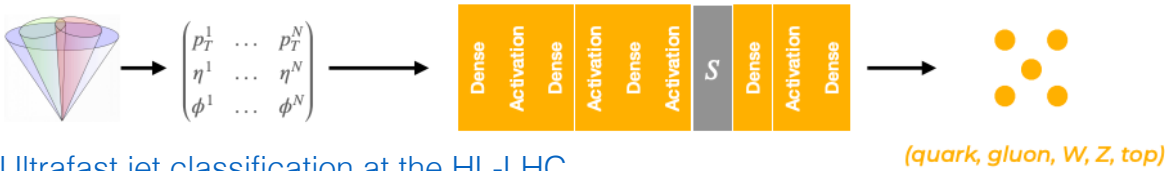
log( $p_T$ )  
record filled  
PID: photon  
ElectroMagnetic ID  
mass  
relative  $p_T$   
PID: muon +  
 $d_{xy}$   
PID: muon -  
PID: hadron +  
PID: hadron neutral  
PID: hadron -  
PUPPI Weight  
PID: electron +  
 $p_T$   
PID: electron -  
Track Quality  
 $\Delta\phi$   
 $\Delta\eta$   
 $z_0$



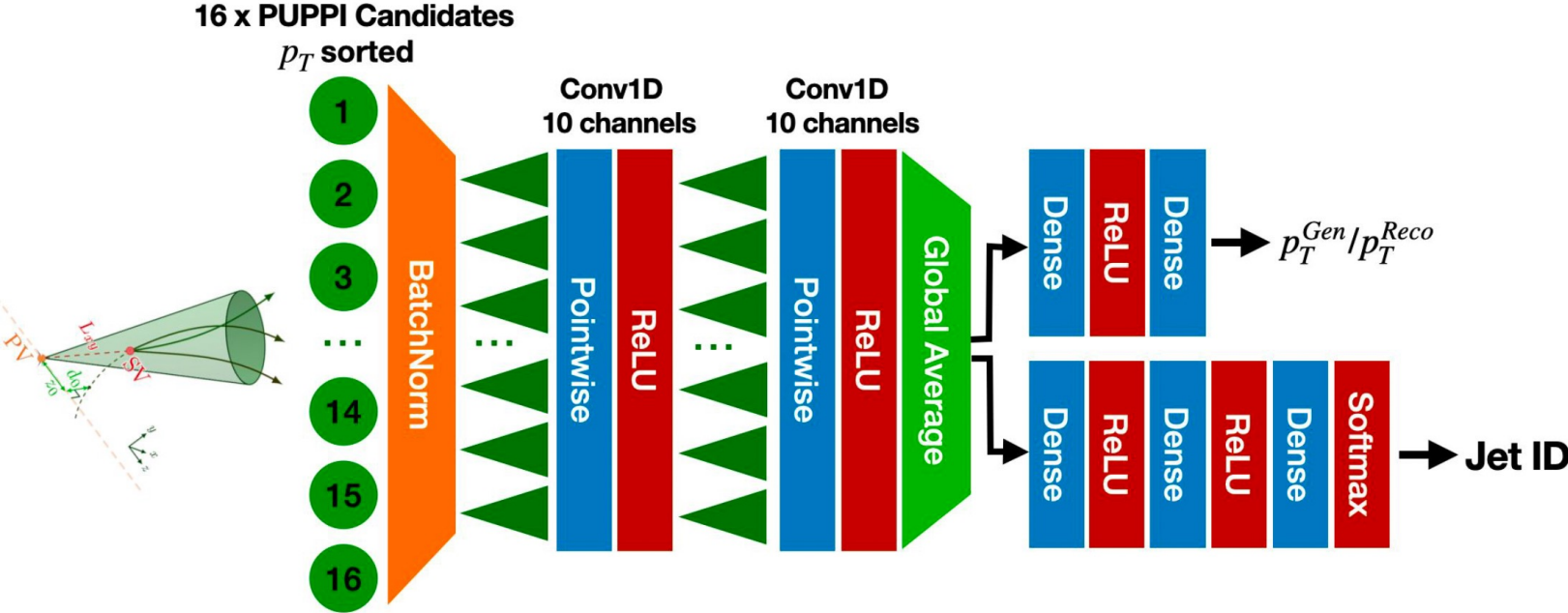
# One network

DeepSets architecture,  
lightweight and powerful

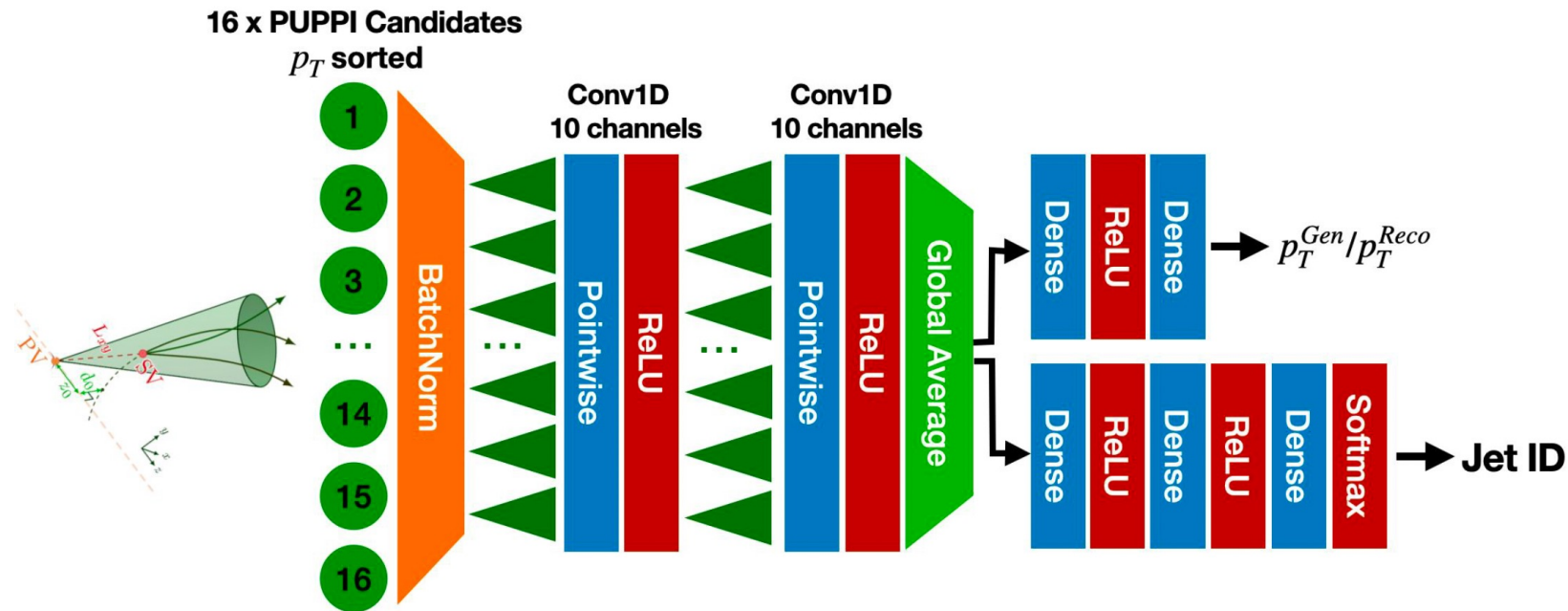
## Deep Sets DS



[Ultrafast jet classification at the HL-LHC](#)

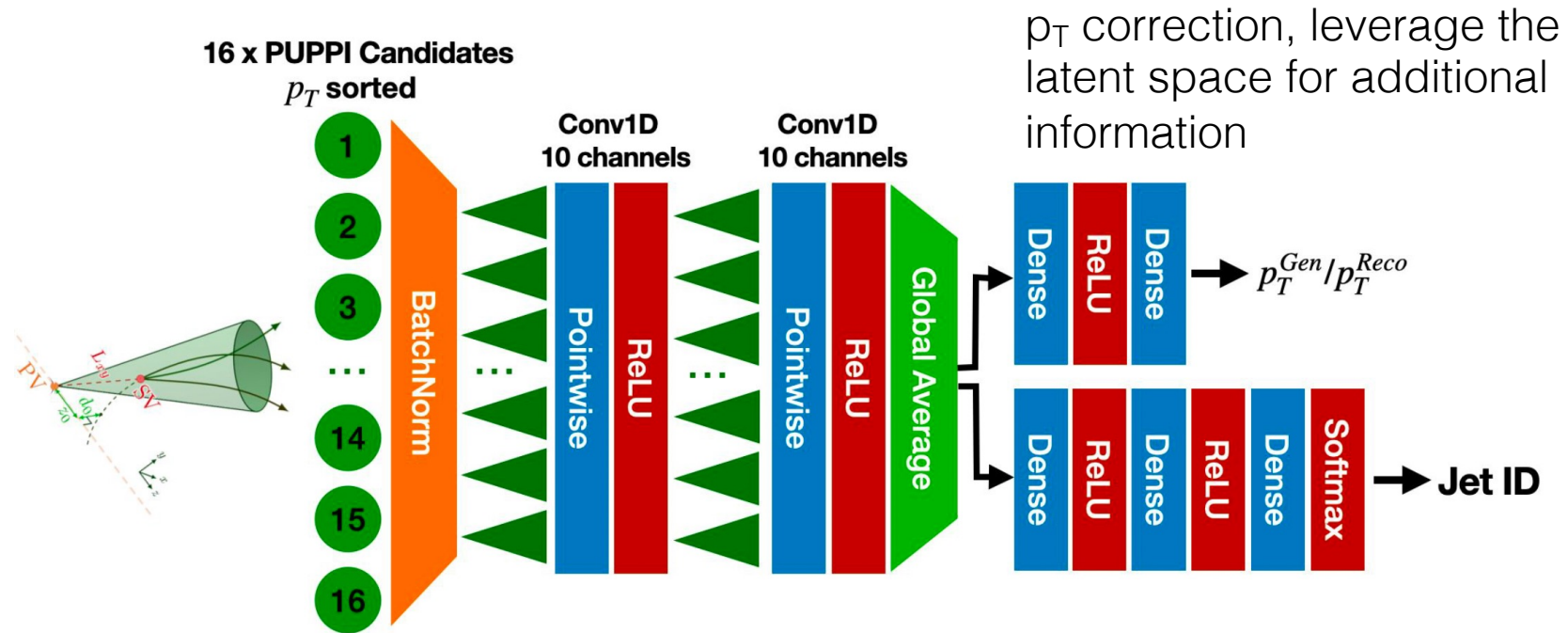


# One network

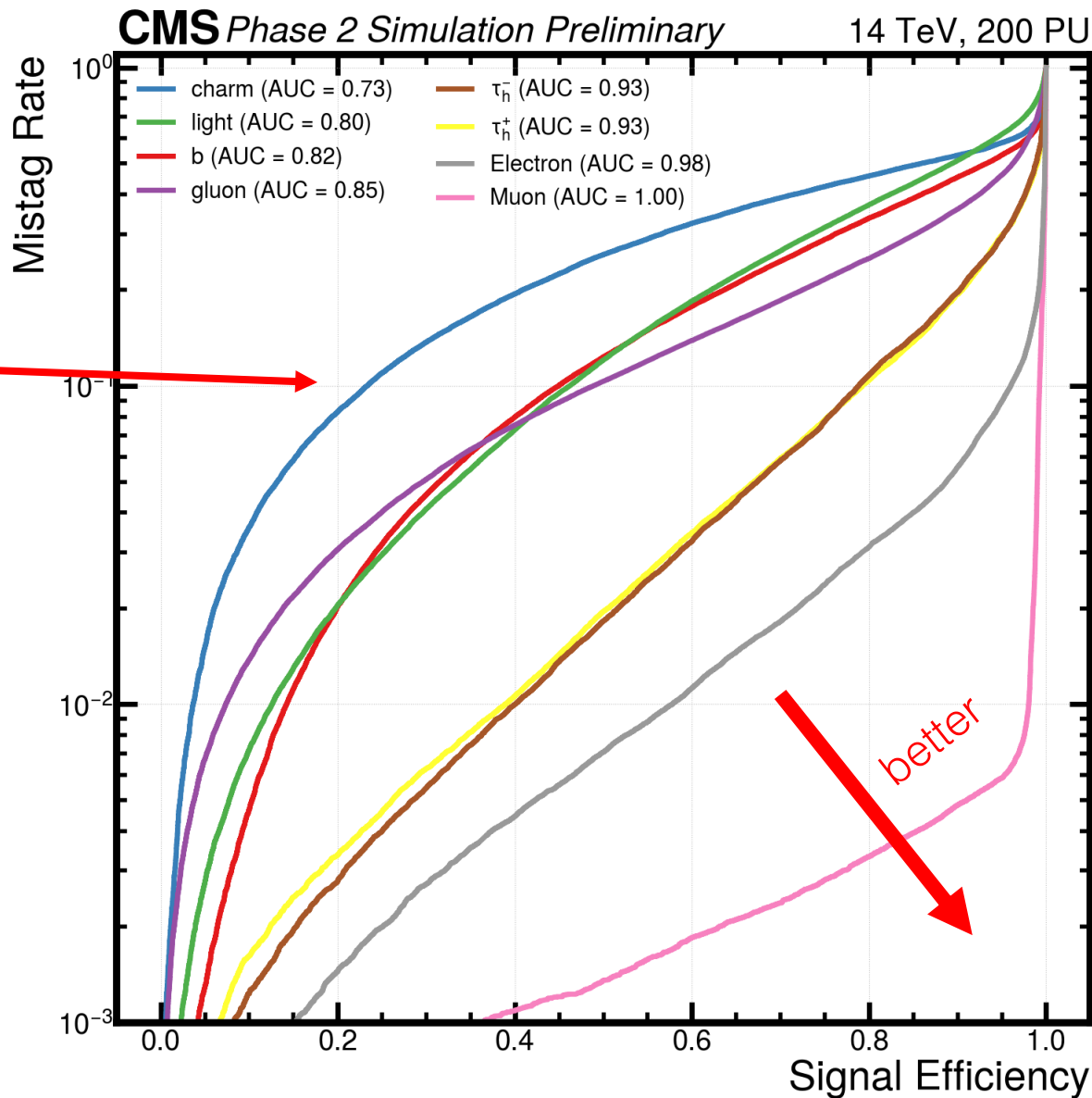
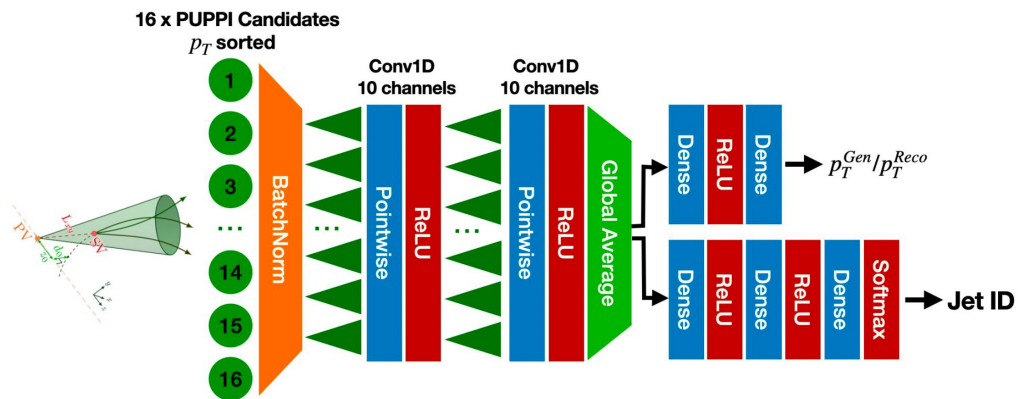


Jet class ID, 8 classes to choose from

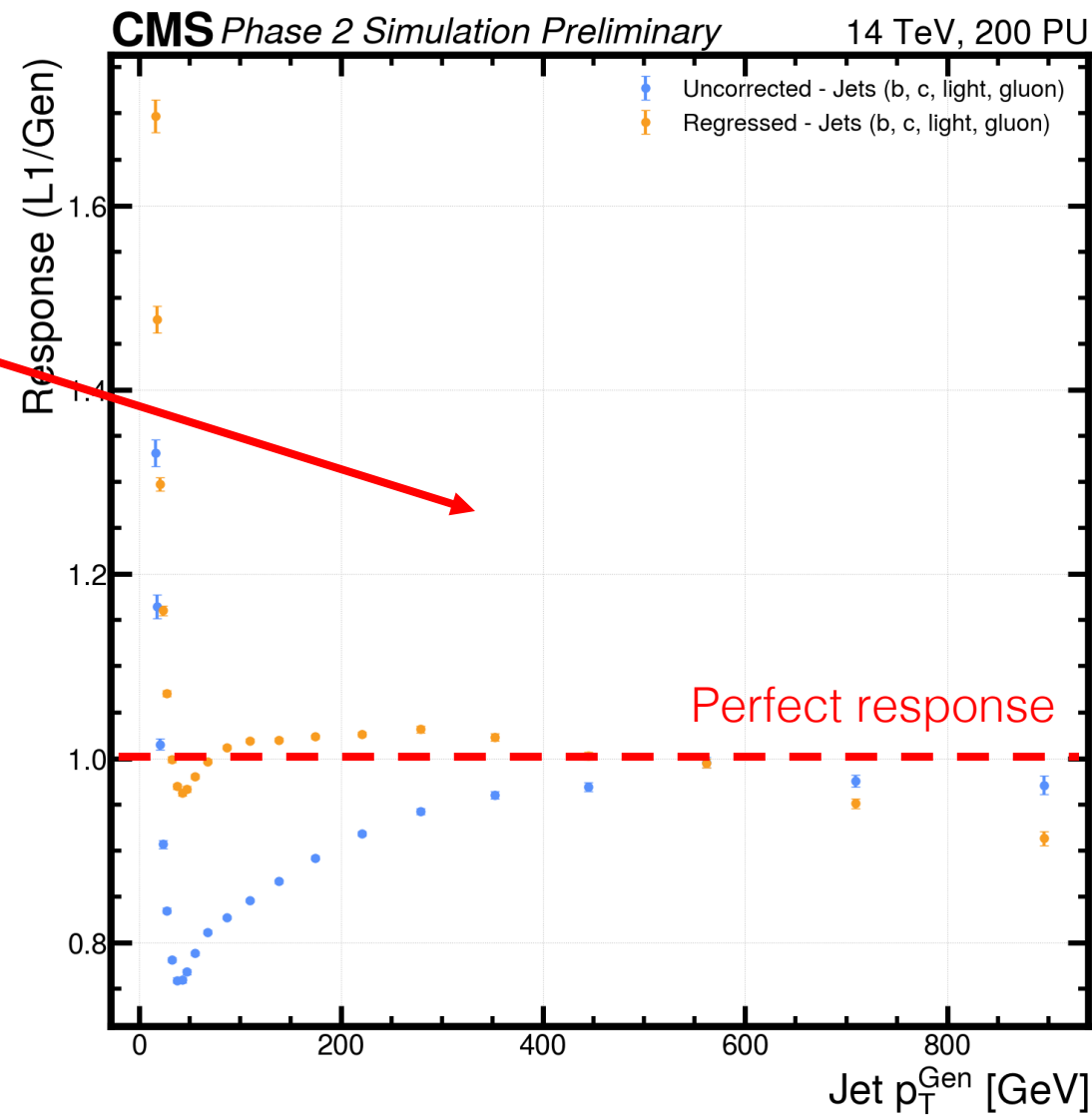
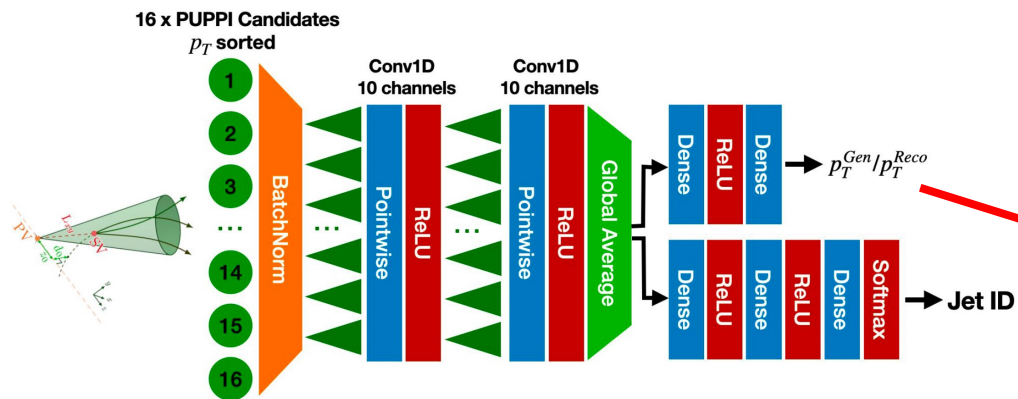
# One network



# One DeepSets network, but how does it perform?

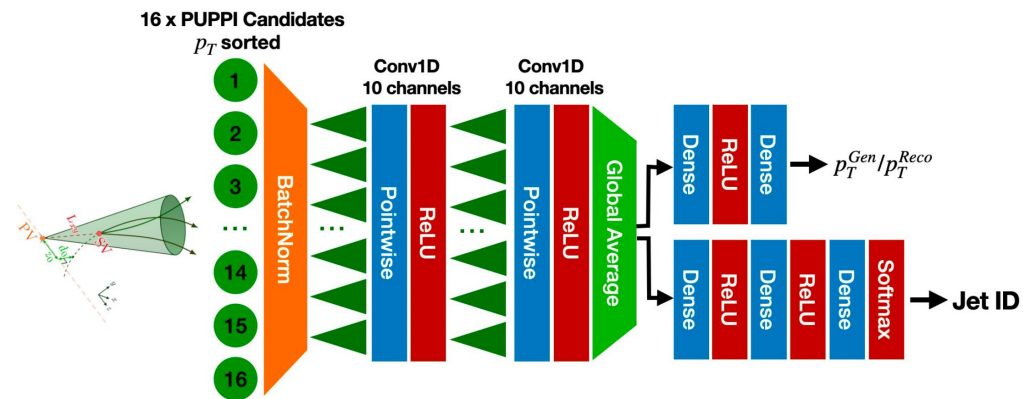


# One DeepSets network, but how does it perform?

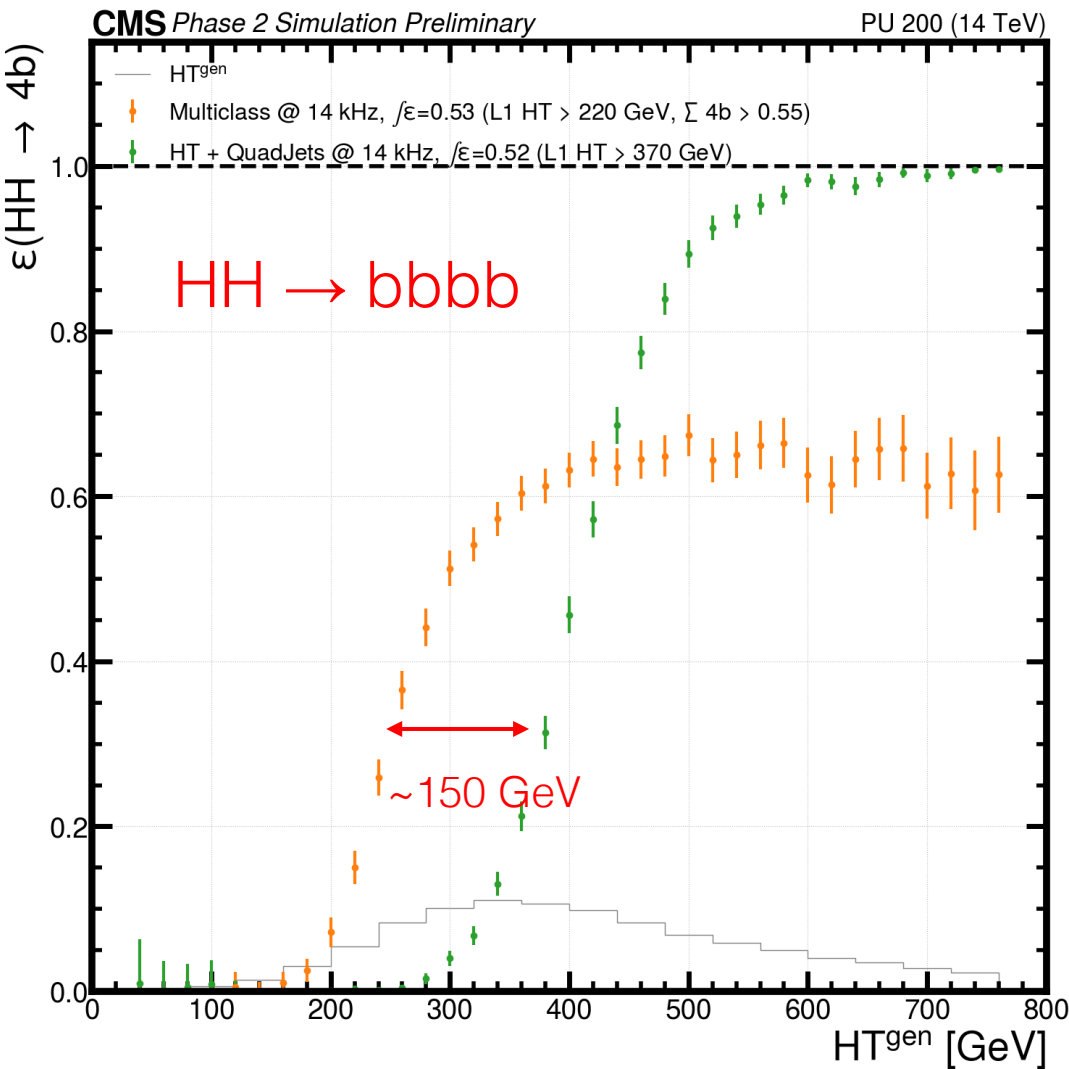




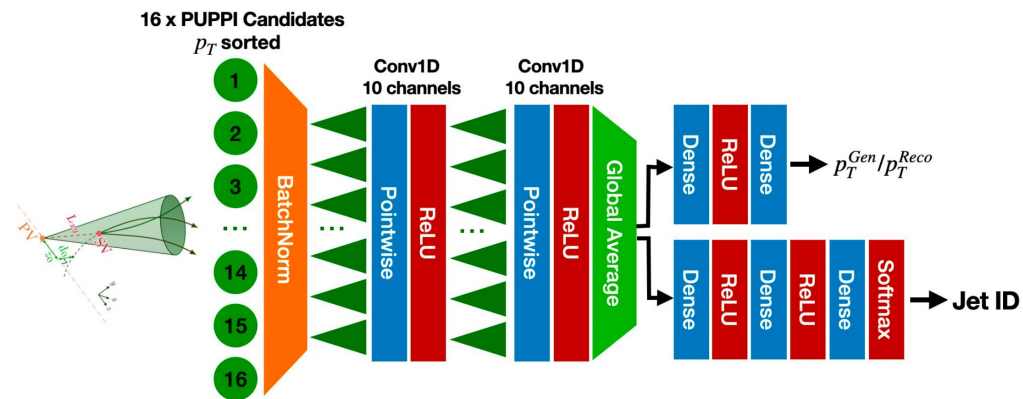
# One DeepSets network, but what about the physics?



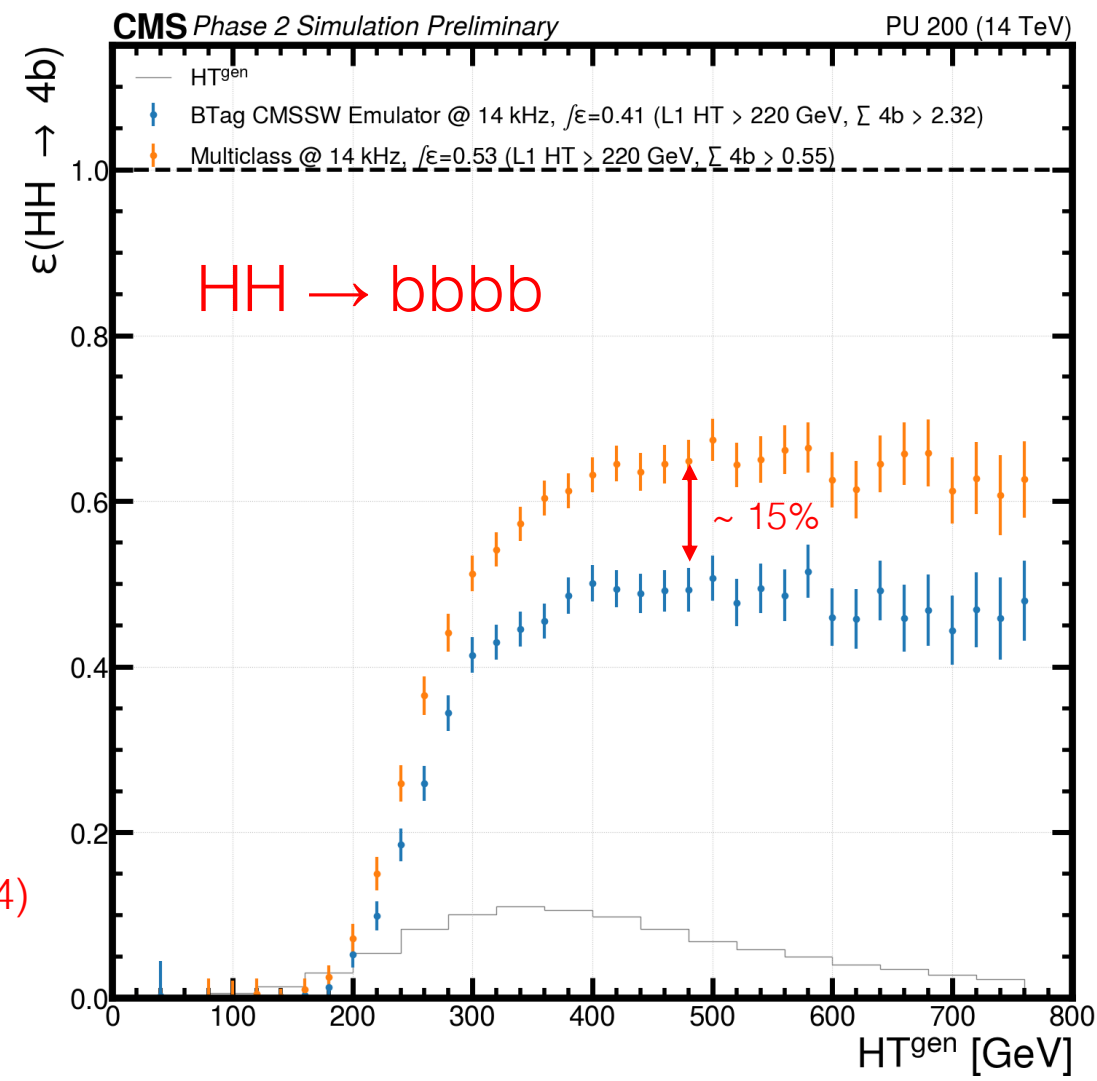
DeepSets Tagger vs  $H_T + 4$  jets trigger



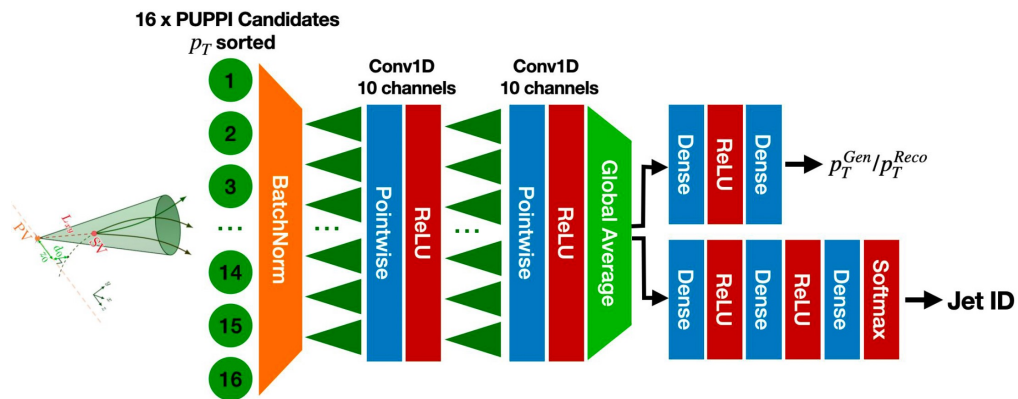
# One DeepSets network, but what about the physics?



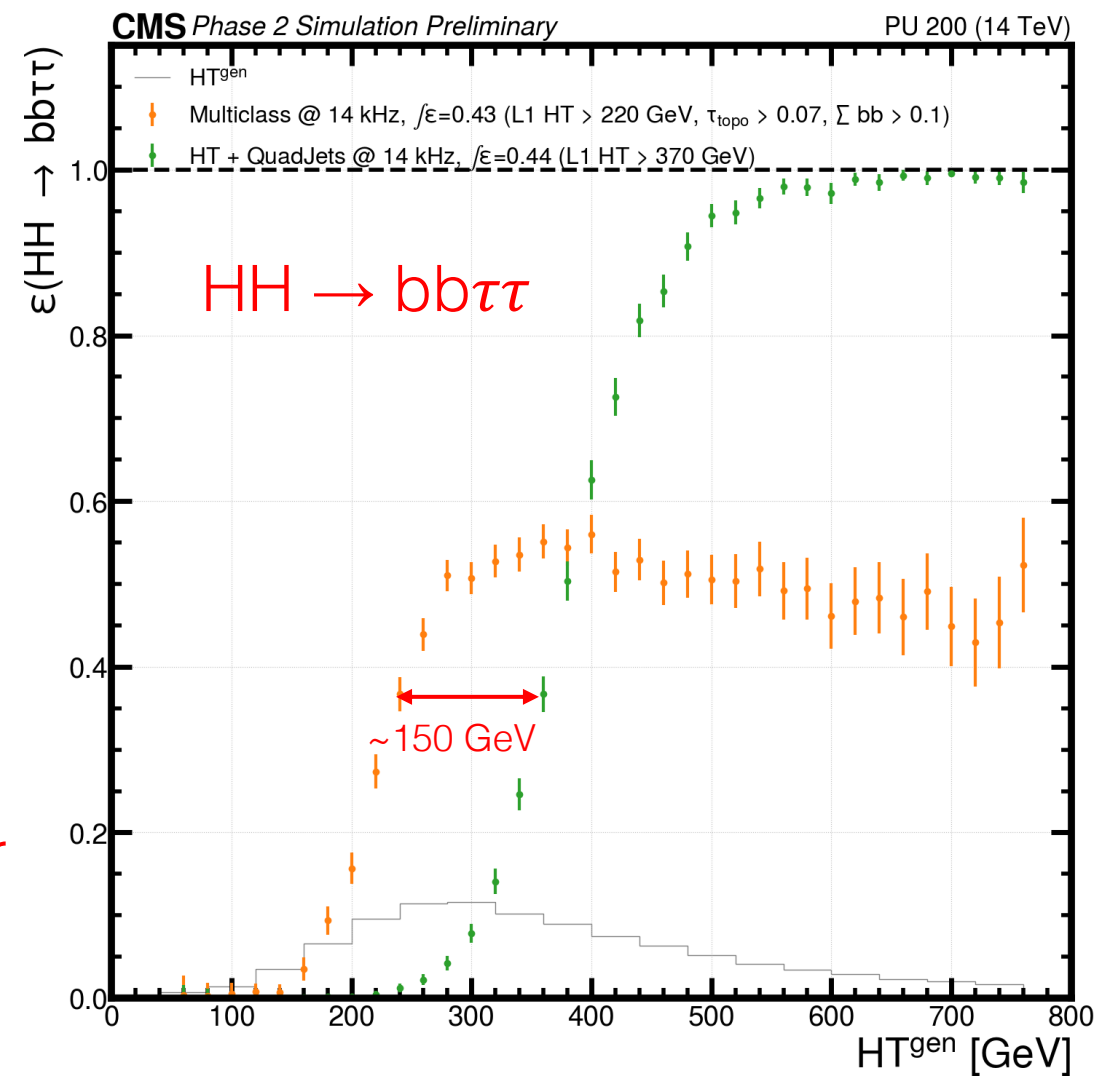
DeepSet Tagger vs  
old b jet tagger (slide 14)



# One DeepSets network, but what about the physics?

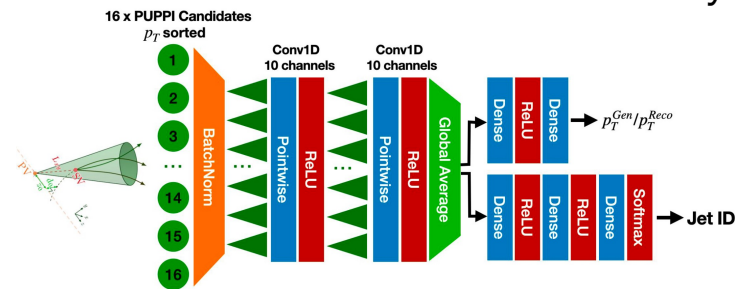


Tagger vs  $H_T$  + 4 jets trigger



# One DeepSets network, what about the firmware?

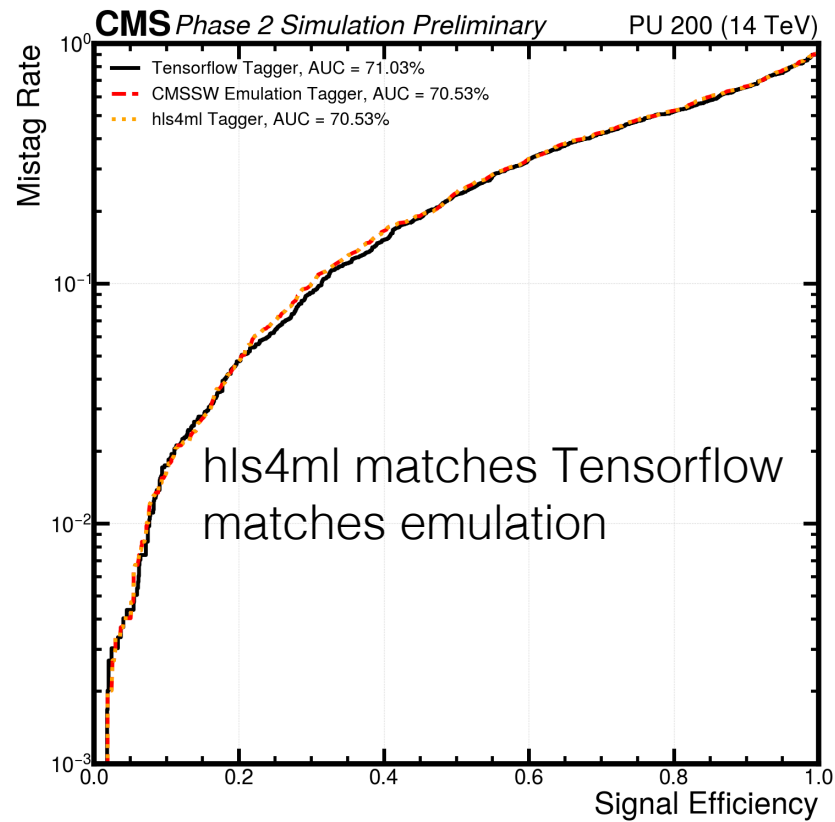
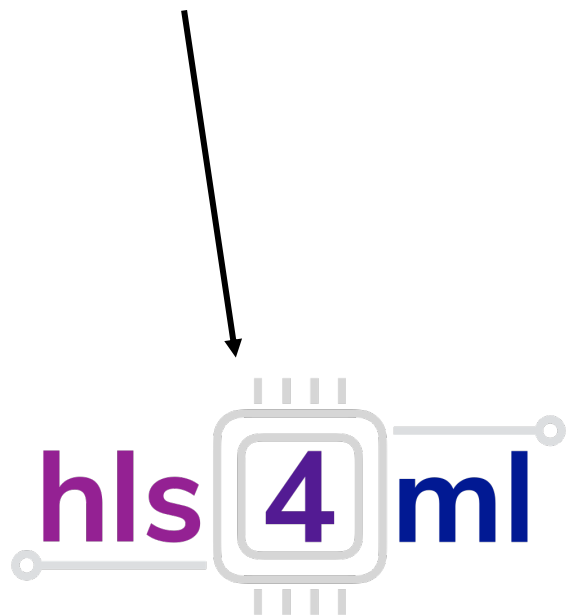
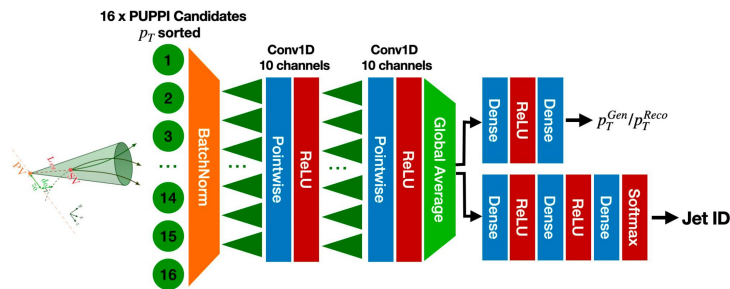
Quantised & Pruned from day 1



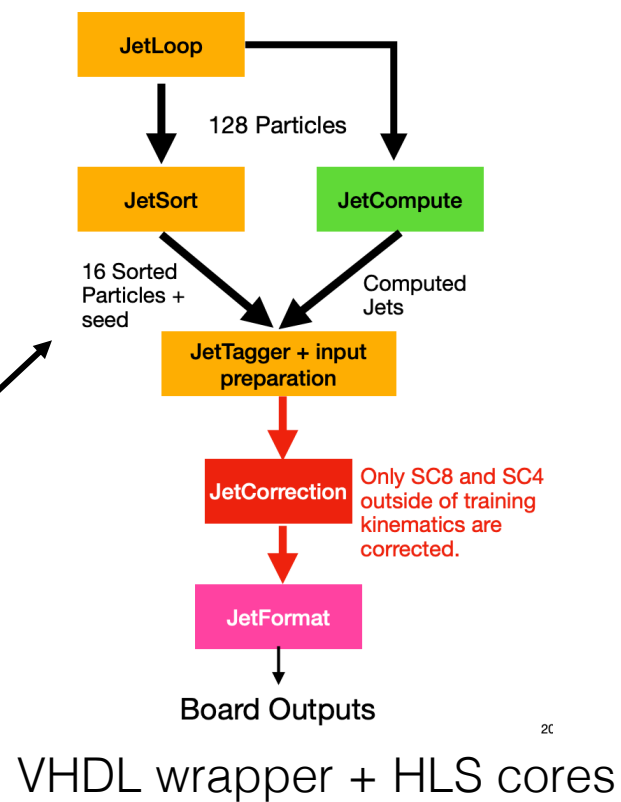
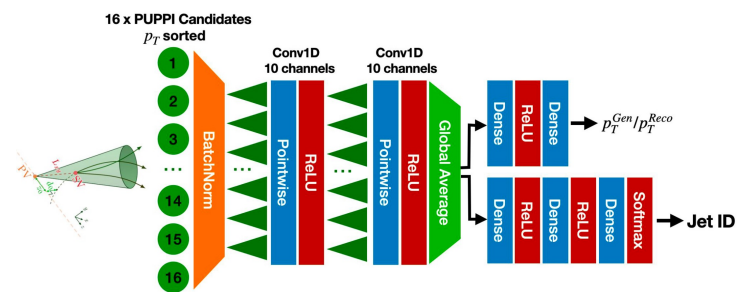
Custom conv1d implementation



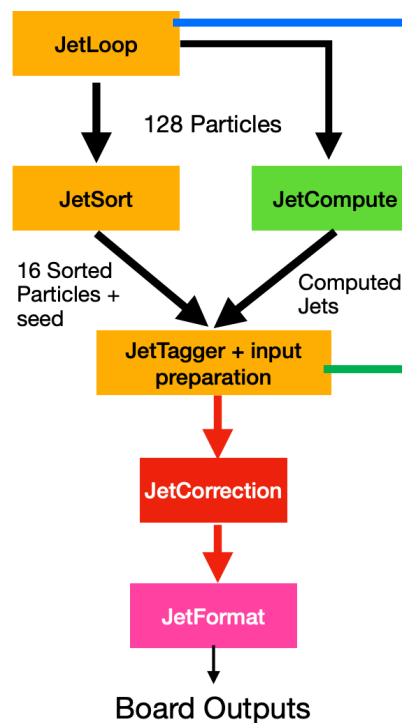
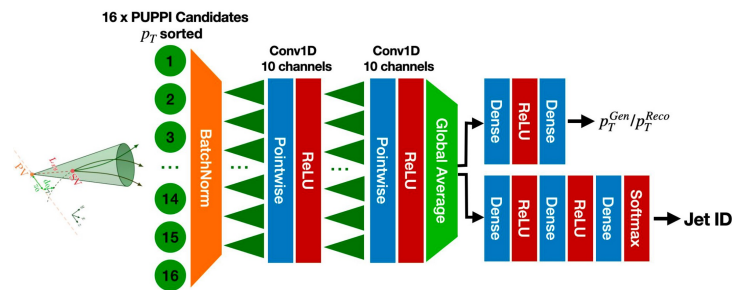
# One DeepSets network, what about the firmware?



# One DeepSets network, what about the firmware?



# One DeepSets network, what about the firmware?

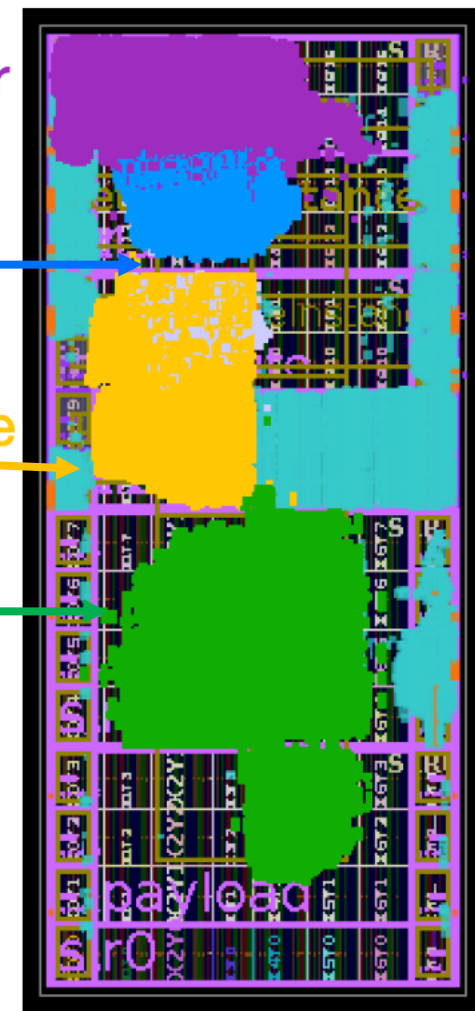


Deregionizer

Jet Loop

Jet Compute

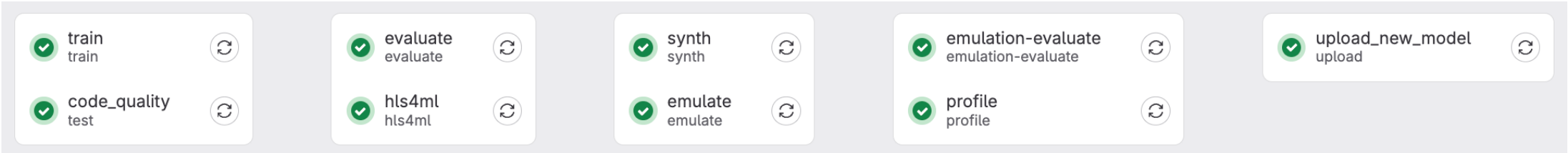
Jet Tagger



VU13P 360 MHz 1 SLR for Jet Tagger  
II = 1 cc  
Latency = 80 cc including 31 for input  
prep. 18 cc spare for improvements



# One DeepSets network, what about the deployment?



# One DeepSets network, what about the deployment?

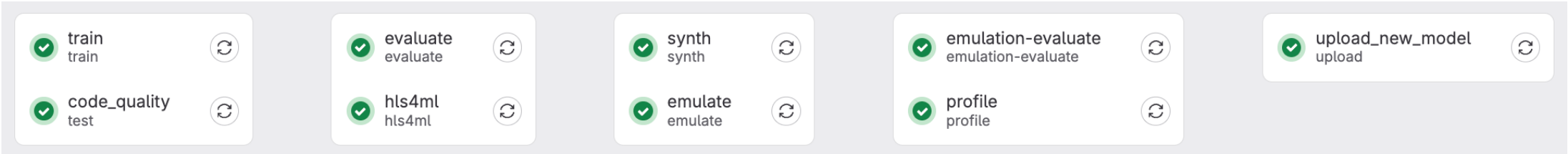
Generic training  
block, put in  
whatever  
architecture you like

Create ~100  
physics plots

Build hls model  
using CERN  
ci4fpga service

Check the emulation  
output and compare  
to hls4ml and python

Upload the model,  
plots and firmware  
to a website registry



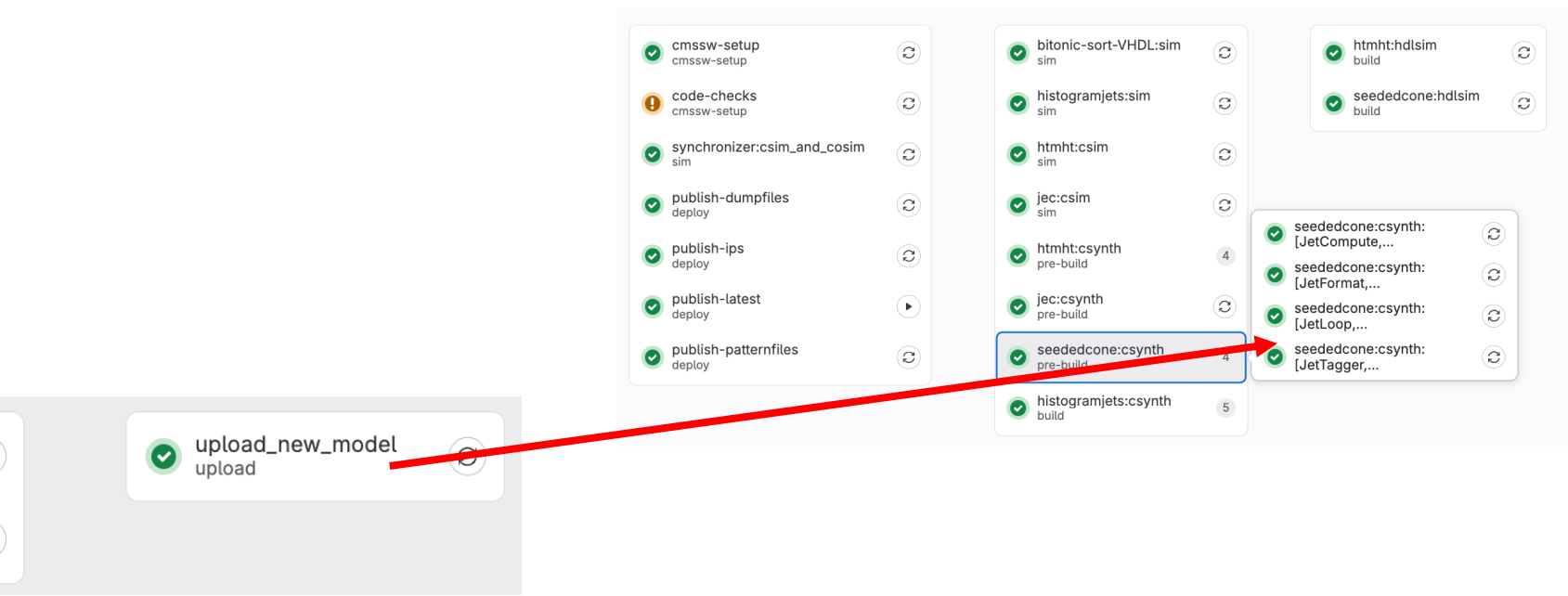
hls4ml create  
project

Run new model  
in cmssw for bit  
accurate  
emulation

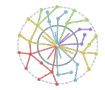
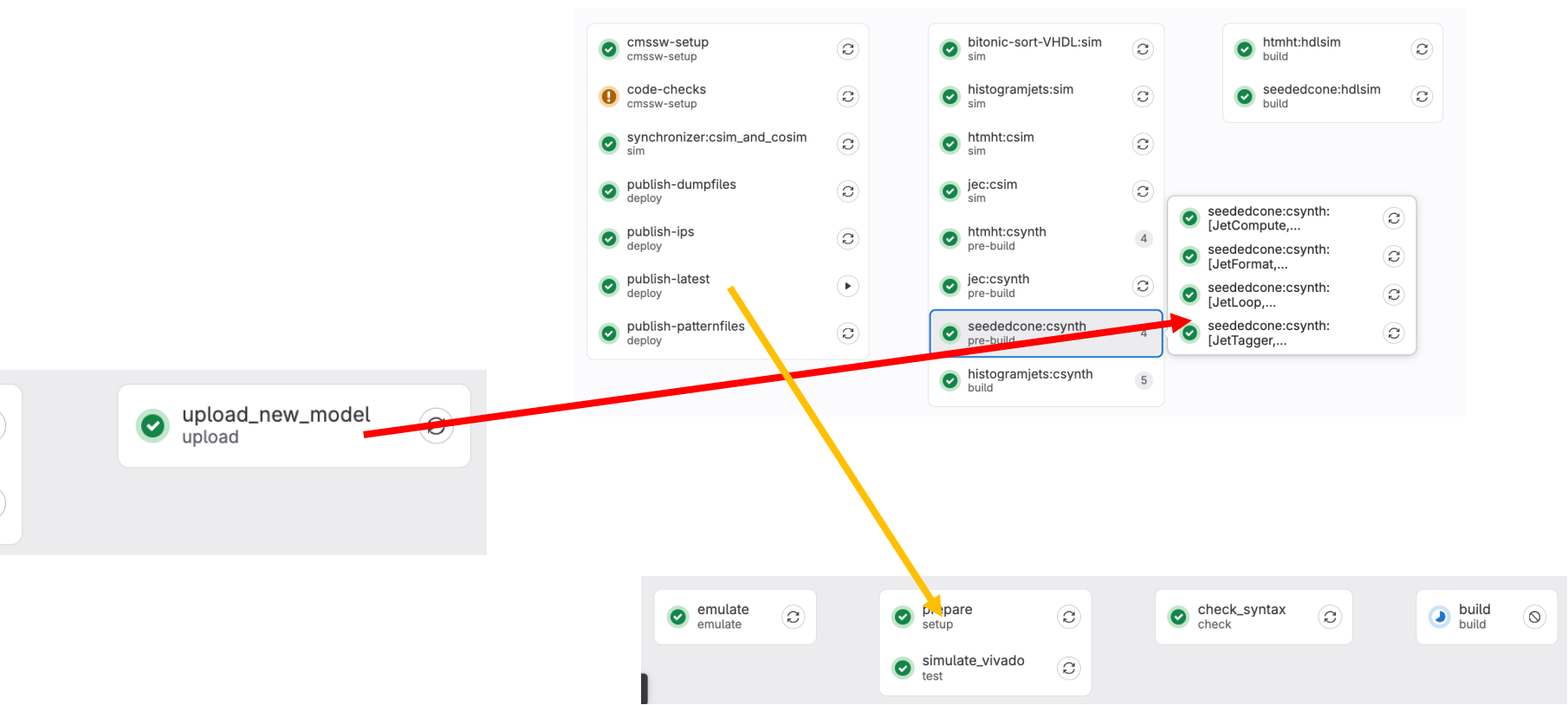
Profile the  
hls4ml, check  
the resources



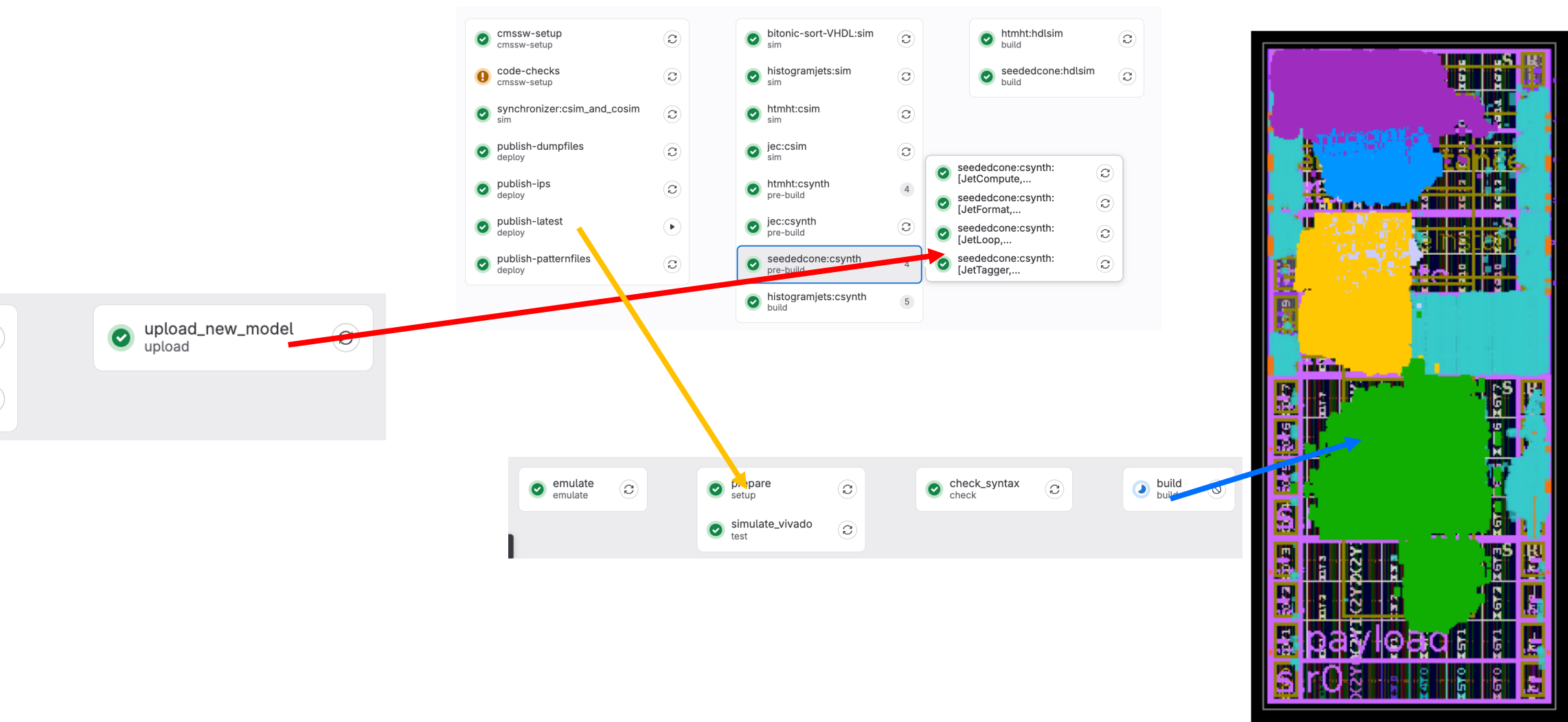
# One DeepSets network, what about the deployment?



# One DeepSets network, what about the deployment?



# One DeepSets network, what about the deployment?



# One DeepSets network, what about the deployment?

Every main branch model will get:

- A trained model ✓
- A suite of physics performance plots ✓
- A bit accurate emulator ✓
- A fully profiled hls4ml firmware project ✓
- A bit file of the trigger board we target (ongoing.....)

Without a single command being typed by the user  
(apart from git push)



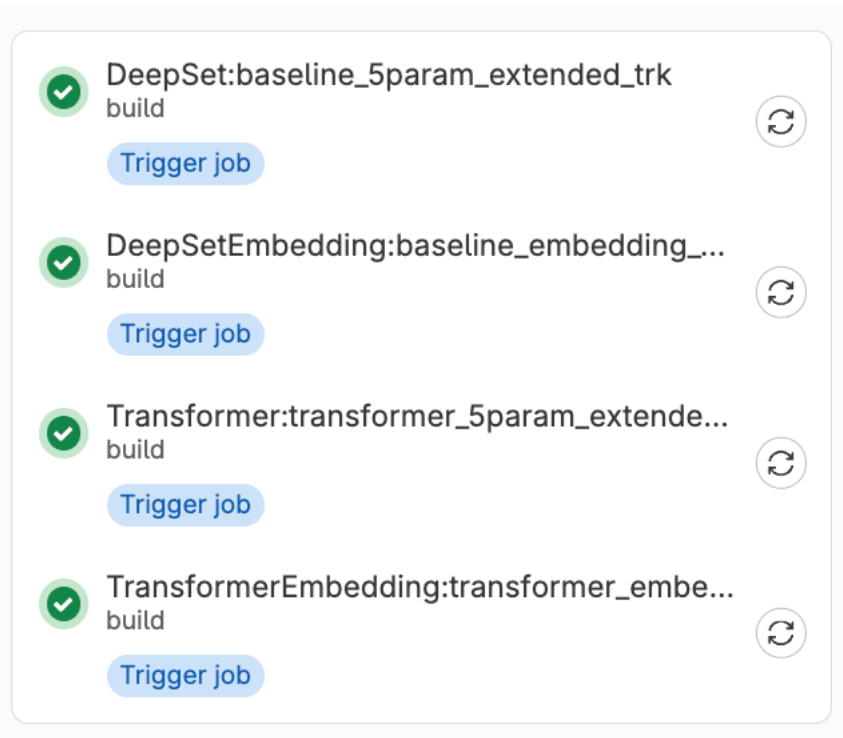
# One DeepSets network, what about the deployment?

Every main branch model will get:

- A trained model
- A suite of physics performance plots
- A bit accurate emulator
- A fully profiled hls4ml firmware project
- A bit file of the trigger board we target

Without a single command being typed by the user  
(apart from git push)

Aim to be fully  
generic, not limited to  
DeepSet  
architectures....



.... or even the jet tagging, automatic  
workflows for all FPGA ML projects ....

MLops [BoF session](#) on  
thursday



# What's next?

## Firmware

- Model running in **hardware** tested and **matching emulation**
- **HGQ, HGQ2, da4ml** models all ready to be fully CI integrated

## Models

- **Transformer** models, **contrastive learning embedding** models, **interaction-net** model, **JEDI-net** models, **Vector tree** models ....

## Robustness

- Jet tagging models in **changing environments**, initial studies on **Lipschitz networks**, **adversarial training** upcoming

## Physics

- **Jet level features**, how to use the scores, **event topology taggers**, **new jet classes** to add and test





# Backup



