Short exercise: Trigger

Deborah Pinna, Victor Shang, Ganesh Parida, Pampa Goose
(University of Wisconsin Madison)

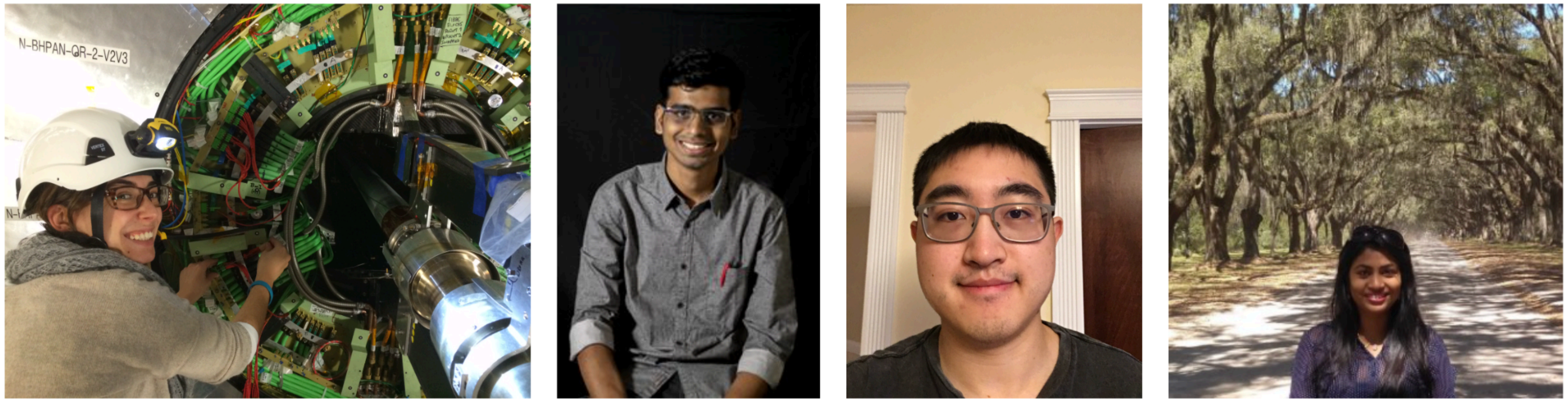
Virtual CMSDAS@LPC 2022

4-7 January 2022

Slides based on previous ones from S. Laurila (CMSDAS 2021)

Welcome

Trigger facilitator team



Deborah Pinna | **Ganesh Parida** | **Victor Shang** | **Pampa Ghose**

Exercises

- **Goals**
  - introduction to trigger ***basic concepts and terminology*** (e.g. turn-on, prescale, matching...)
  - learn about ***main features of the CMS trigger system***
  - ***overview of the CMS trigger menu***, allowing to identify suitable triggers for physics analysis
  - overview of analysis tools needed to ***access trigger-related information*** in CMS datasets
  - perform ***trigger efficiency measurements***
- If something is unclear or doesn’t work, we are here to help! Don’t hesitate to ***ask questions and discuss*** in the dedicated Mattermost channel
  - *https://mattermost.web.cern.ch/cmsdaslpc2022/channels/shortextrigger*

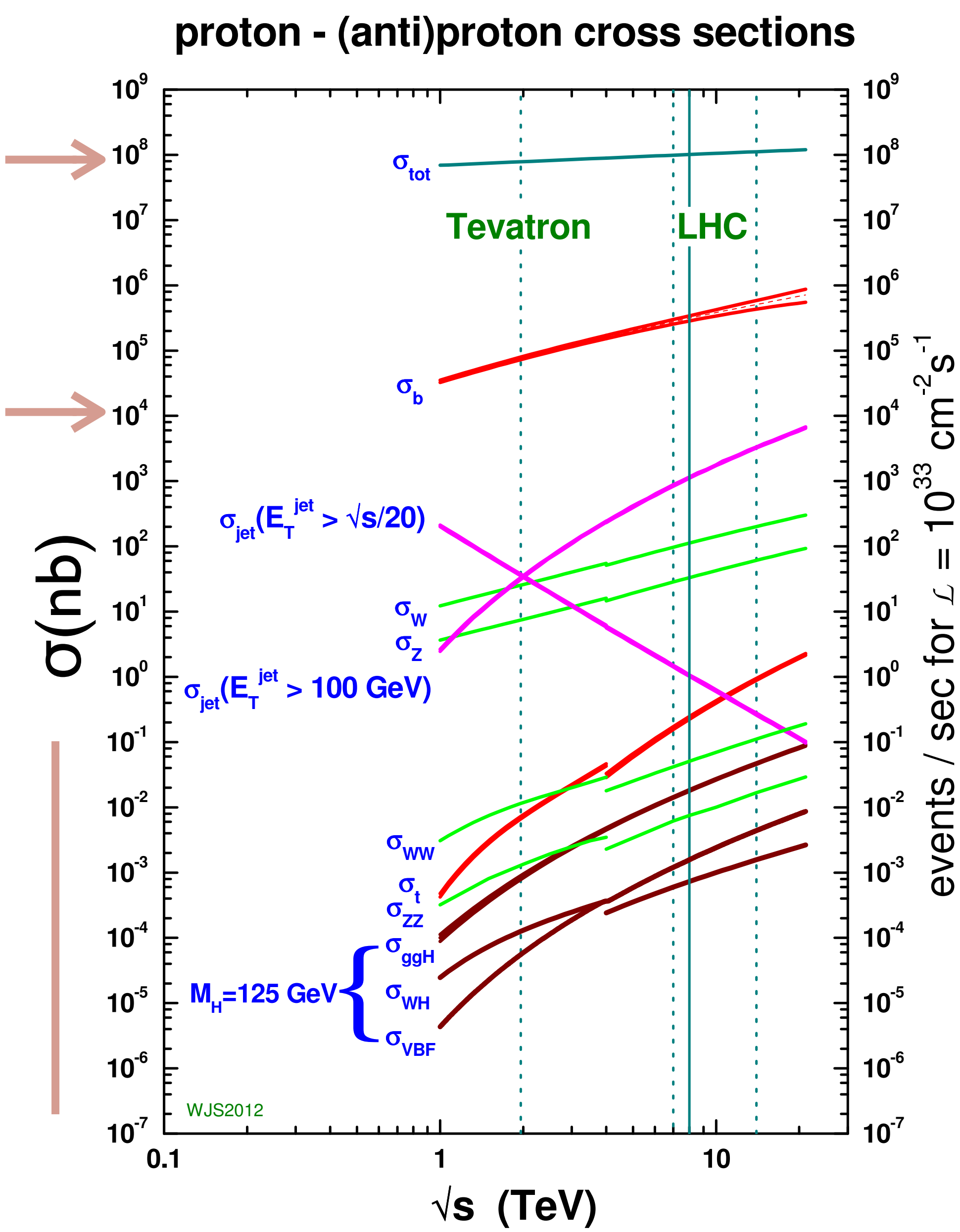
Why we need a trigger?

- ***40 MHz bunch crossing***
  - translates in ~1MB/events and tens of terabytes of data per second
  - to reduce the amount of events to a level that we can read out, store and process offline we need a trigger system!
- ***With a ~1 kHz output rate we can have a rich physics program***
  - ***but*** we need be able to select the “interesting” events and reject ~99.998% of all the data
- ***Trigger performance is critical***
  - we can only analyze what the trigger system has accepted
  - if an event is missed there is no way to recover it

LHC collision rate: 40 MHz

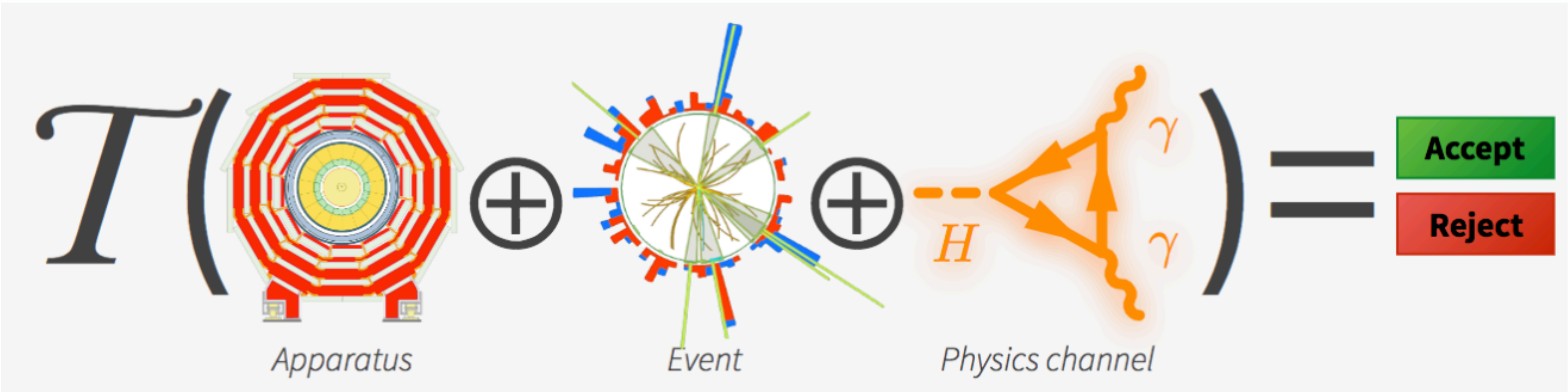
Offline storage rate: 1 kHz

Typical “interesting” physics processes: < 10 Hz



Trigger importance in your analysis

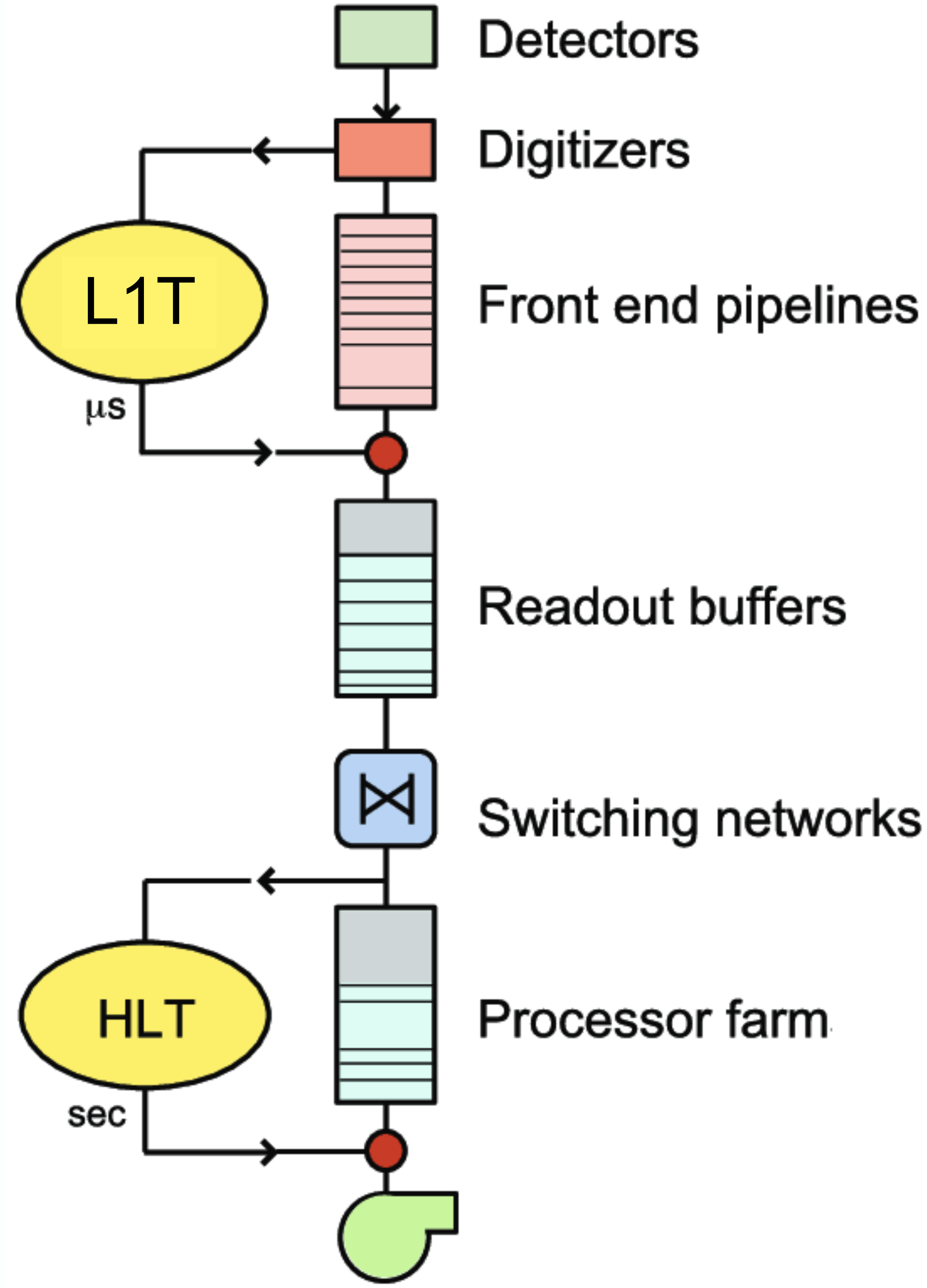
- Every ***CMS analysis is limited by the data the trigger system*** has collected
  - the CMS trigger system defines the physics reach
- When ***planning a new analysis*** we need to consider:
  - a trigger algorithms that can select the signal of interest does already exist?
  - do we need to design such algorithms and wait for the next run?
- If a trigger of interest for our analysis exists:
  - need to ***measure the trigger efficiency,*** or if already calculated, is it applicable in your case?
  - some corrections and/or syst. uncertainties are needed
  - ***in Exercise 4 you will learn how to perform an efficiency measurement***
- Investigating what happened in the trigger when it collected your data, you can get ideas for improving the current trigger algorithms for future runs, or designing completely new ones!



Overview of CMS trigger system

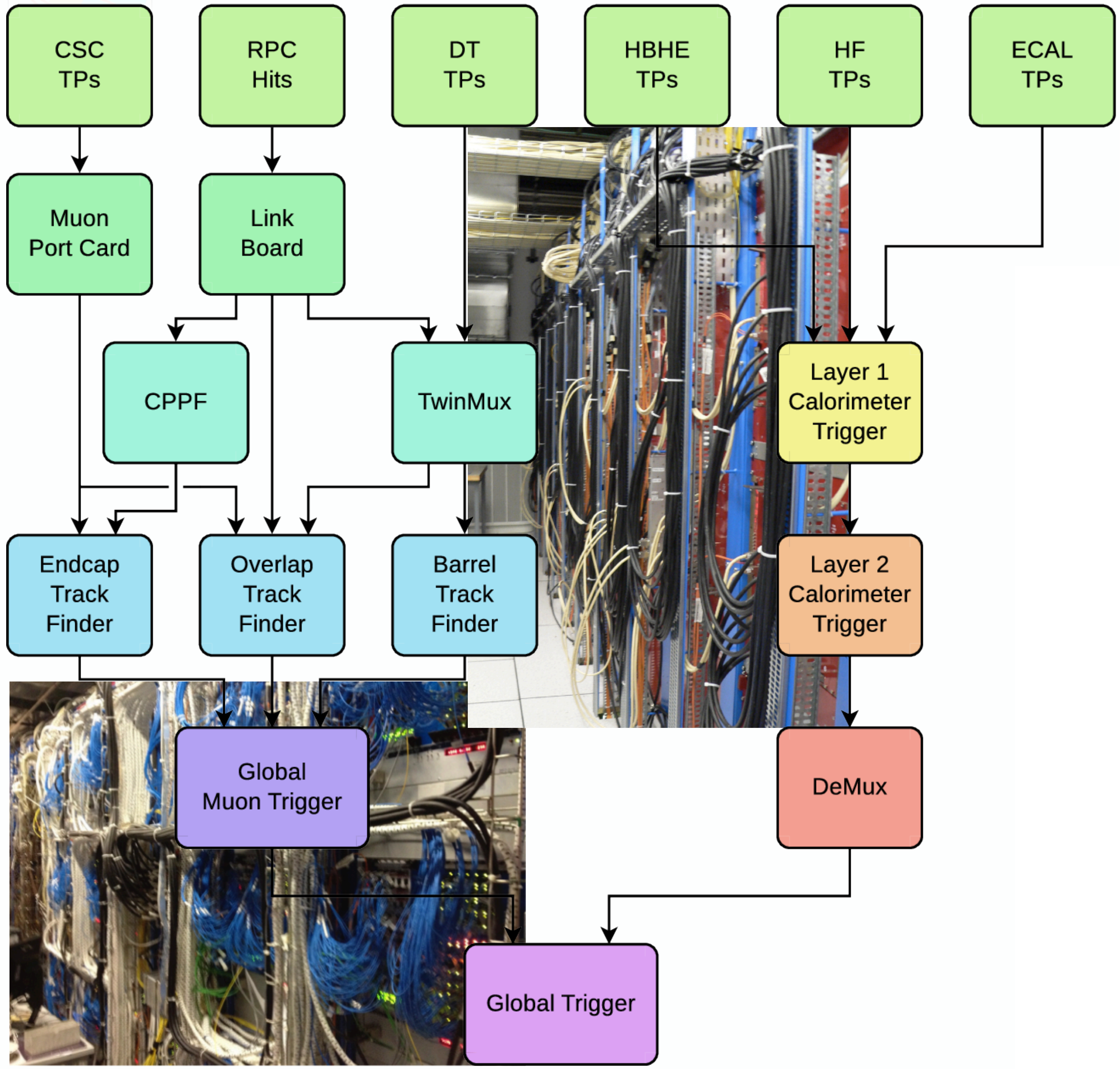
- ***Level-1 Trigger (L1T)***
  - purely ***hardware-based***
  - custom-made chips (ASICs) and programmable logic (FPGAs)
  - decision in **~4 μs** (hard limit)
  - 40 MHz down to **100 kHz** (hard limit)

- ***High Level Trigger (HLT)***
  - CPU farm with ~30k cores processing events in parallel
  - 100 kHz down to ~1 kHz (soft limit)
  - decision in ~100 ms (soft limit)



Level-1 (L1T) trigger structure

- Uses ***inputs only from calorimeters and muon systems***
- Signals pre-processed into trigger primitives (TPs) with limited granularity
- Modular architecture: different TPs from same event are processed in parallel, until ***Global Trigger*** gets the ”full picture” and makes the final ”L1Accept” decision based on trigger menu



Level-1 trigger menu

- ***List of active trigger algorithms*** (with specific selection thresholds) used to select events is called ***trigger menu***

- At L1T, the menu consists of ***trigger seeds***, such as:

L1 Trigger — *Selection/thresholds*

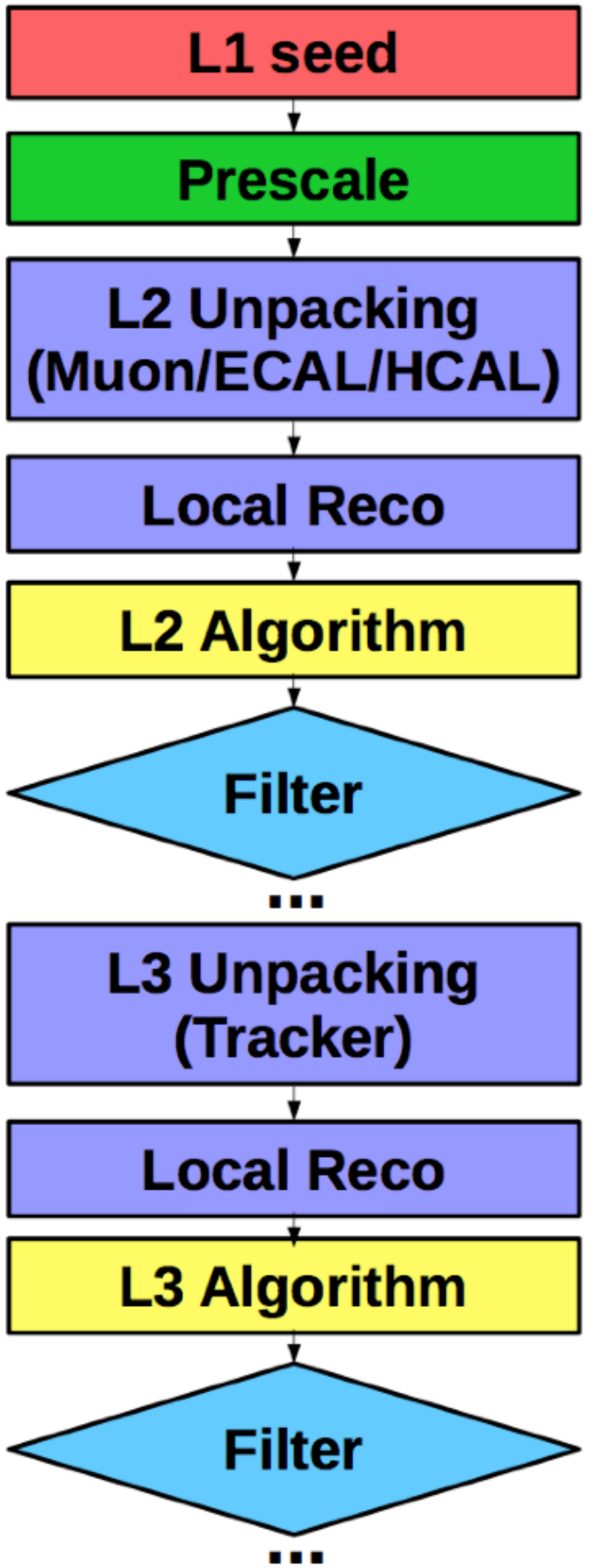
Algorithm name	Description
L1_SingleLooseIsoEG28er2p5	Single loosely isolated e/γ with $E_T > 28$ GeV and $\lvert\eta\rvert < 2.5$
L1_DoubleIsoTau32er21	Double isolated τ with $E_T > 32$ GeV and $\lvert\eta\rvert < 2.1$
L1_SingleMu22	Single muon with $p_T > 22$ GeV
L1_DoubleEG_25_12_er2p5	Double e/γ with $E_T > 25$ GeV, 12 GeV and $\lvert\eta\rvert < 2.5$
L1_DoubleMu_15_7	Double muon with $p_T > 15$ GeV, 7 GeV
L1_ETMHF100	$E_T^{miss} > 100$ GeV
L1_SingleJet180	Single jet with $E_T > 180$ GeV
L1_DoubleJet150er2p5	Double jet with $E_T > 150$ GeV and $\lvert\eta\rvert < 2.5$

- To avoid ***dead time***, L1T menu is adjusted so that the combined rate from all active seeds stays below 100 kHz

- Rate of seed can be reduced by ***prescaling***:
  - **eg.** prescale of 3 means that only one in 3 events that pass all selection criteria actually fire the trigger → rate reduced to 1/3

High Level Trigger (HLT) structure

- ***HLT receives full event information*** (including tracker)
- A **L1 seed** sets off algorithms for given HLT paths
  - each **HLT path** targets a physics object (or a combination) passing specific selections *(e.g.HLT_IsoMu19_eta2p1_LooseIsoPFTau20)*
- **Prescales** can be applied also at HLT
- **HLT objects** are **reconstructed** with CMSSW software similar to offline reconstruction, but optimized to be ~100x faster
  - same modules can be used by different HLT paths
  - run fast processing steps (**L2**) first and slower ones (**L3**) later
  - **Filters** to reject event and stop processing as soon as possible
- **HLT objects are similar to offline objects** (often looser selections are used), and they are stored together with HLT decisions (1/0) for each HLT path



High Level Trigger (HLT) menu

- During Run 2, a typical ***HLT menu*** contained ~400 HLT paths
  - any event that passes at least one path is stored in one of the primary datasets (e.g. SingleMuon, JetHT, DoubleEG...)
- All HLT paths active in Run 2 and their primary datasets are listed in a twiki page: https://twiki.cern.ch/twiki/bin/viewauth/CMS/HLTPathsRunIIList

2018

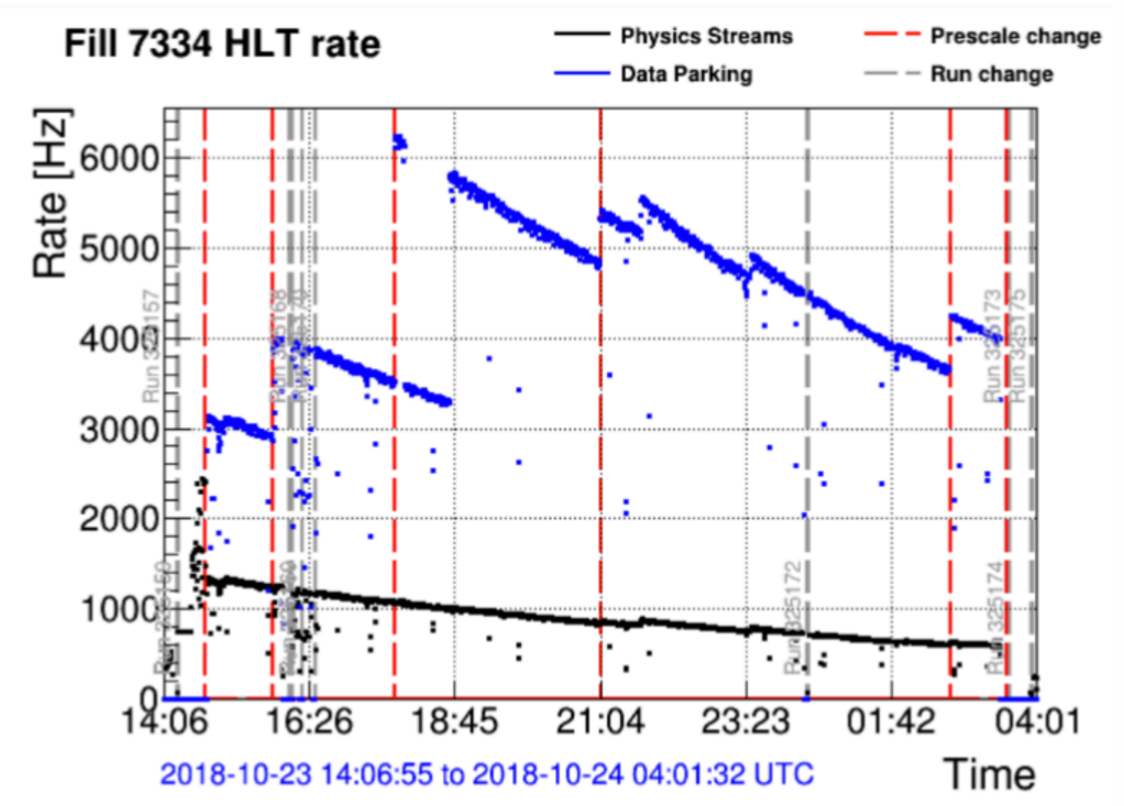
path	act. lumi (fb-1)	eff. lumi (fb-1)	first run	last run	menu version	dataset
HLT_BTagMu_AK4DiJet110_Mu5_noalgo_v	31.92	4.18	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK4DiJet110_Mu5_v	28.04	2.14	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK4DiJet170_Mu5_noalgo_v	31.92	20.31	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK4DiJet170_Mu5_v	28.04	11.71	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK4DiJet20_Mu5_noalgo_v	31.92	0.0329	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK4DiJet20_Mu5_v	28.04	0.0213	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK4DiJet40_Mu5_noalgo_v	31.92	0.21	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK4DiJet40_Mu5_v	28.04	0.13	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK4DiJet70_Mu5_noalgo_v	31.92	1.01	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK4DiJet70_Mu5_v	28.04	0.64	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK4Jet300_Mu5_noalgo_v	31.92	31.92	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK4Jet300_Mu5_v	28.04	28.04	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK8DiJet170_Mu5_noalgo_v	31.92	16.64	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK8DiJet170_Mu5_v	28.04	9.44	315257	320065	v1.1	BTagMu
HLT_BTagMu_AK8Jet170_DoubleMu5_noalgo_v	31.92	31.92	320673	325172	v3.3	BTagMu
HLT_BTagMu_AK8Jet170_DoubleMu5_v	22.74	22.74	315974	320065	v2.0	BTagMu
HLT_BTagMu_AK8Jet300_Mu5_noalgo_v	31.92	31.92	320673	325172	v3.3	BTagMu

Data-taking techniques: data parking

- ***Prescaling*** can give us access to lower-p_T objects while reducing the trigger rate
    - ***but*** with prescale N we only collect 1/Nth of the data
    - is there any other way for data-taking?

- ***HLT rate constraint*** rises from two sources:
    - storage rate
    - offline reconstruction capacity

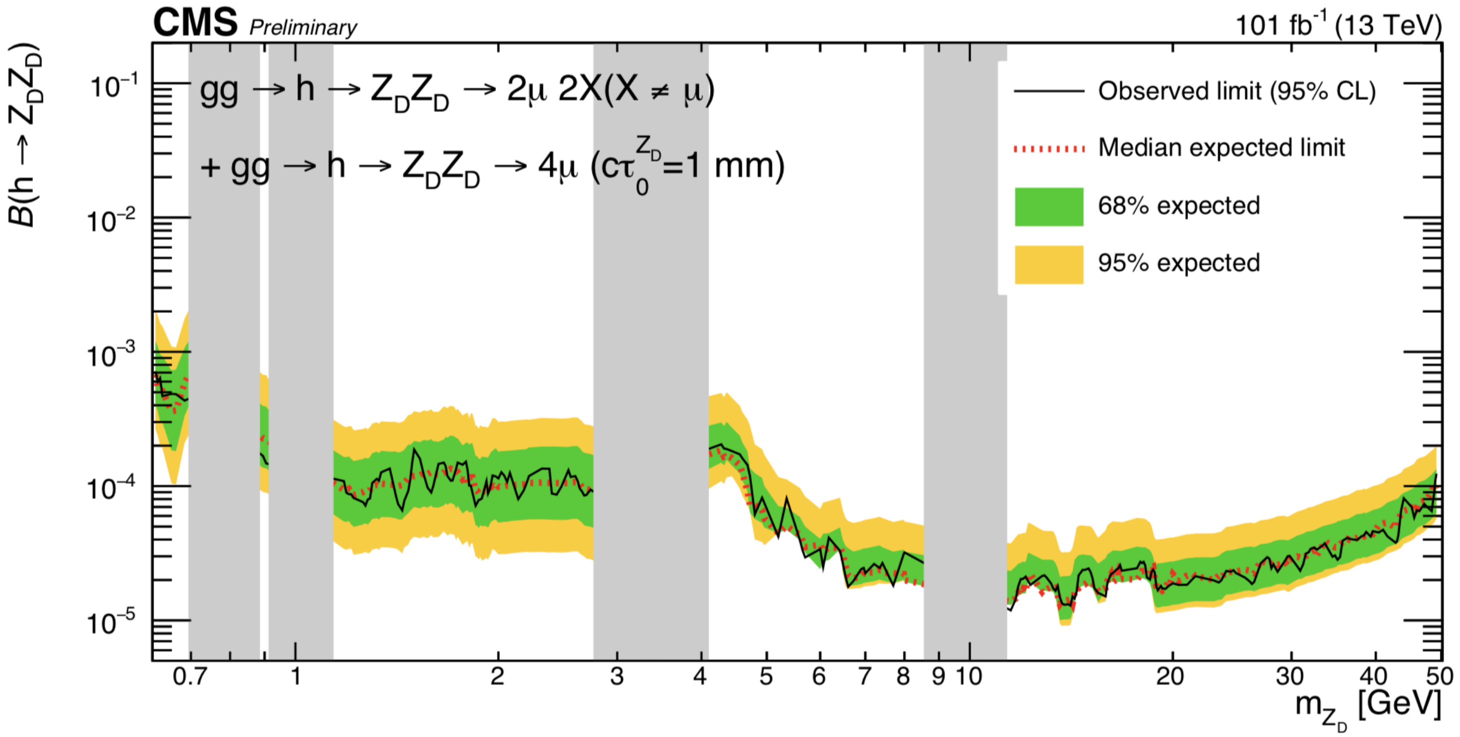
- ***Data parking*** is a data-taking technique that reduces offline reconstruction needs
    - ***record data to disk at a higher rate, but postpone offline processing*** to future (until experiment is no longer running, e.g. during shutdown periods)
    - eg. CMS employed this in 2018 to record more data for B physics studies



Data-taking techniques: data scouting

- ***Data scouting*** circumvents both the storage rate and the offline processing constraints
  - ***idea:***
    smaller data per event to be stored, events at higher rates can be stored and process within constraints
  - in practice:
    store only information of objects reconstructed at the HLT
    → lower trigger thresholds, higher acceptance of low-p_T physics processes
  - ***caveats:***
    - need to get the reconstruction (and calibrations!) right at first try
    - resolution of your physics objects is limited → not suitable for all analyses

di-muon low-mass resonance search (EXO-20-014)



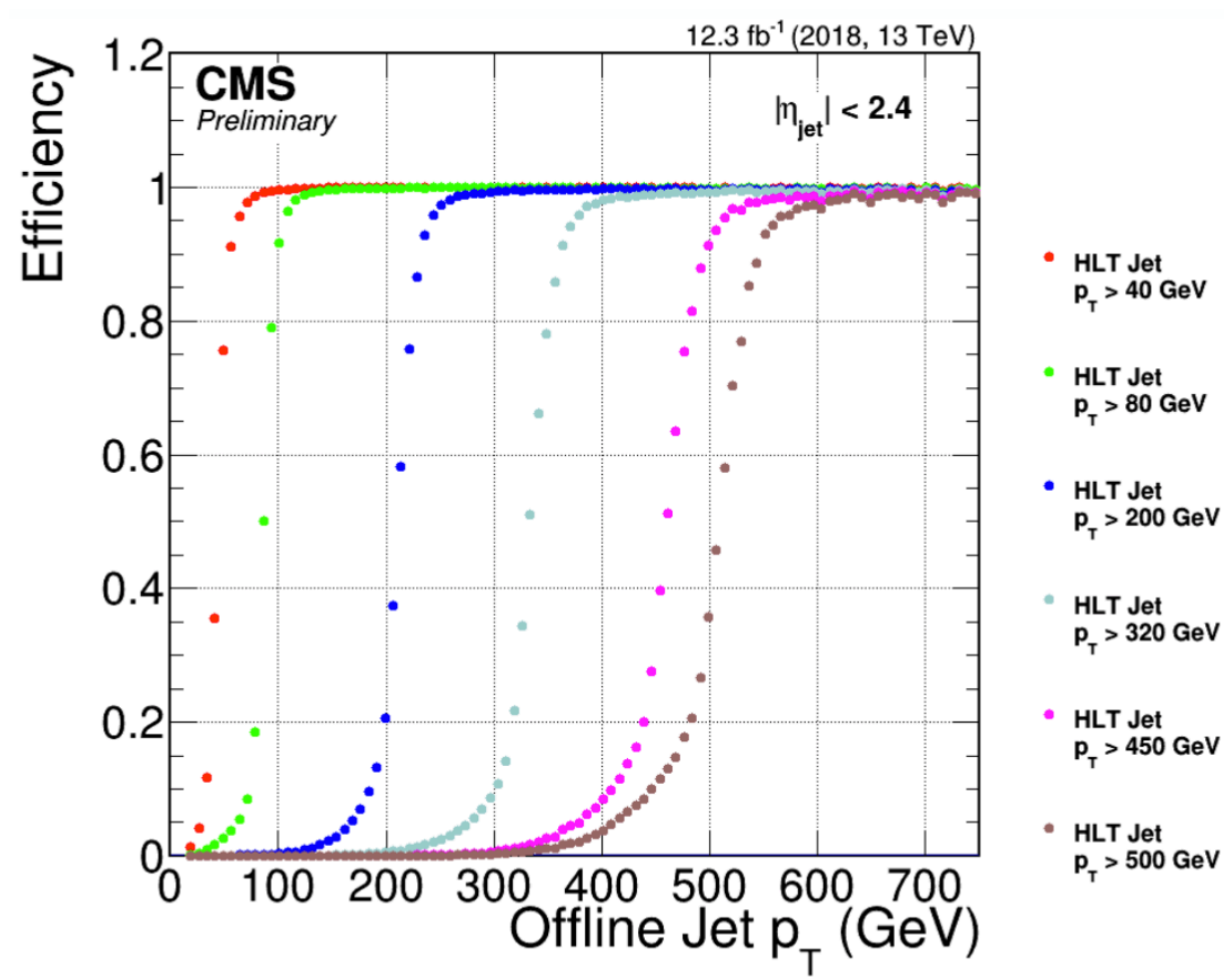
Trigger efficiency

- **Trigger efficiency**
  - fraction of events the trigger was supposed to collect vs faction really collected: **efficiency = # events that passed trigger / # all events**

- ***For an analysis***, we need to understand how the trigger efficiency depends on different "offline" variables

- Typically, efficiency plots show a **turn-on** region due to limited p_T resolution at trigger level, followed by a **plateau**

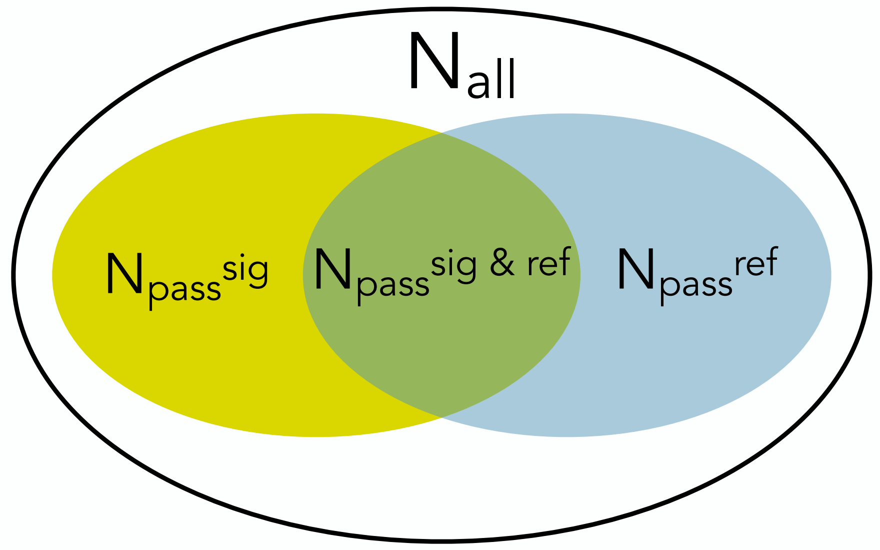
- ***How can we measure this efficiency?***



Trigger efficiency: with a reference trigger

- With ***simulated events***, we can calculate the efficiency $\varepsilon = N_{pass}/N_{all}$
- How can we figure out the efficiency of a specific trigger ***in data***, when we don't have access to N_{all}?
  - we can use a **reference trigger**, which fires **independently** of the trigger of interest (signal trigger)
  - then the probability for an event to pass both signal and reference triggers is $\varepsilon_{sig\&ref} = \varepsilon_{sig} \times \varepsilon_{ref}$

$$\rightarrow \ \varepsilon_{sig} = \varepsilon_{sig\ \&\ ref} / \varepsilon_{ref}$$
$$= (N_{pass}^{sig\ \&\ ref} / N_{all})/(N_{pass}^{ref} / N_{all})$$
$$= N_{pass}^{sig\ \&\ ref} / N_{pass}^{ref}$$



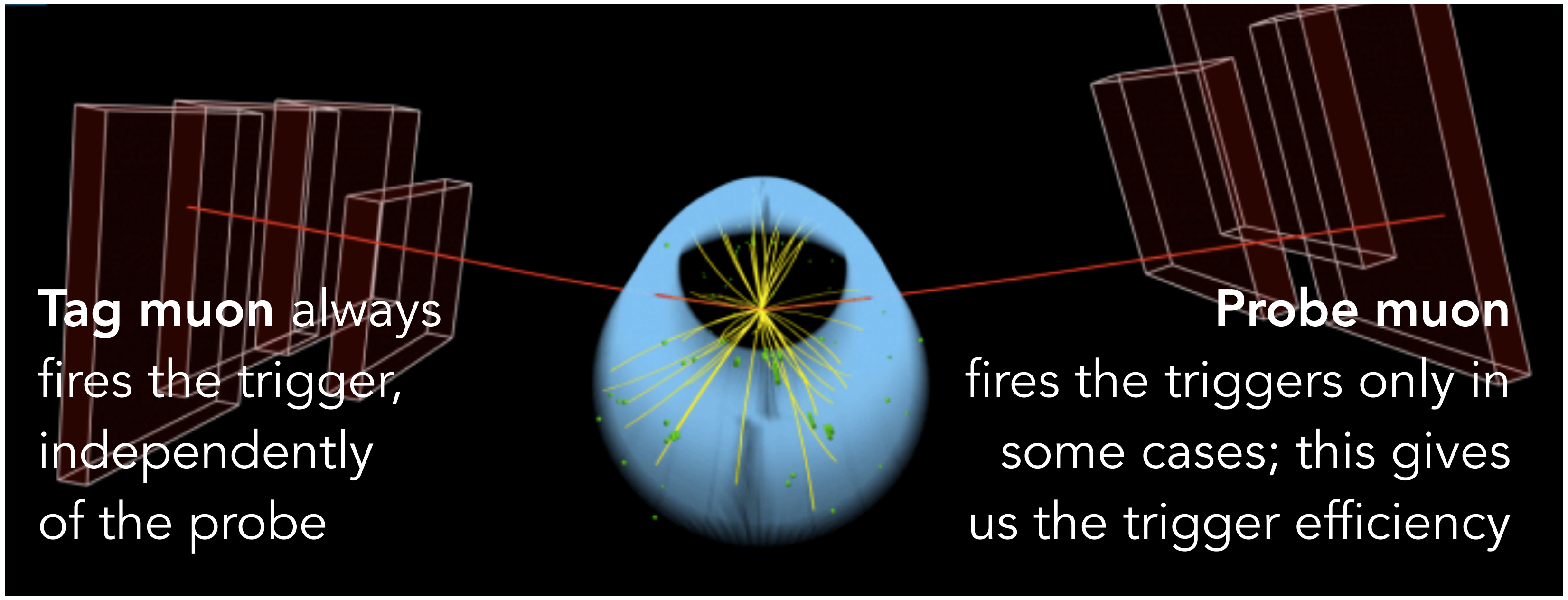
- In ***exercise 1*** you will learn how to do this in practice, using either MiniAOD or NanoAOD format files as input

Trigger efficiency: tag-and-probe method

- In some cases, e.g. with single-muon triggers, we can use the ***tag-and-probe method*** to perform the efficiency measurement ***without a separate reference*** trigger
- In ***exercises 3-4*** we use this method to measure the efficiency of a **single-muon trigger** using dimuon events
    - ***ex. 3***: Access trigger objects in MiniAOD
    - ***ex. 4***: Match trigger objects with the offline objects and calculate the trigger efficiency

$$\varepsilon_{\text{tag\&probe}} = \varepsilon_{\text{tag}}\varepsilon_{\text{probe}}$$

$$\rightarrow \varepsilon_{\text{probe}} = N_{\text{pass}}^{\text{tag\&probe}} / N_{\text{pass}}^{\text{tag}}$$



What's next?

- ***It's time to get started with the exercises!***
  - https://twiki.cern.ch/twiki/bin/viewauth/CMS/SWGuideCMSDataAnalysisSchoolLPC2022TriggerExercise
  - In exercises 1-4 you will see how to perform efficiency measurements in practice
  - there are also a few bonus exercises available if you want to go deeper: for eg. you can learn how to inspect the behavior of the trigger system with OMS (Online Monitoring System)
- In the long exercises you will be using some triggers to select your data – and of course in **your own analysis** as well
- ***Remember***: if something is unclear or doesn't work, we are here to help! Don't hesitate to ***ask questions and discuss*** in the dedicated Mattermost channel
  - *https://mattermost.web.cern.ch/cmsdaslpc2022/channels/shortextrigger*
- During the wrap-up session we can discuss these topics further – feel free to suggest anything that you would like to be covered!

Short exercise Trigger

Now let's start with the exercises!

Backup