

# Table of Contents

<b>Trigger studies</b> .....	<b>1</b>
Overview.....	1
<b>Useful Links</b> .....	<b>2</b>
<b>Trigger rates</b> .....	<b>3</b>
<b>HLT trigger development</b> .....	<b>4</b>
<b>Other useful commands</b> .....	<b>5</b>
Get luminosity.....	5
How to find unrescaled trigger:.....	5
Obtaining pileup rate.....	5
Good Run Lists:.....	5

# Trigger studies

## Overview

This page contains instructions on evaluating trigger rates and developing new trigger paths

On this page:

# Useful Links

- OMS [↗](#) : Online monitor tool to get trigger information

# Trigger rates

Trigger rates for existing triggers can be obtained using OMS [↗](#). For example, if interested in trigger rate of `HLT_HIPFJet140_v1`, first locate useful run (run with larger number of statistics, see here for details)

Navigate CMS → Triggers → HLT Trigger Rates Run:XXXXXX. You can see L! Prerequisites clicking HLT Key menu from CMS → Runs → Run Report Run:XXXXXX

# HLT trigger development

The development is done with [103X\\_dataRun2\\_HLT\\_v1](#) and [CMSSW\\_11\\_3\\_0](#)

In lxplus run the HLT java script:

```
git clone https://github.com/cms-sw/hlt-confdb.git
./hlt-confdb/start
```

Select: [HLTDEVv2](#) and [cms100khz](#) to connect to ConfDB and follow instructions in <https://indico.cern.ch/event/1037579/>

# Other useful commands

## Get luminosity

To get the actual luminosity per trigger use `brilcalc`:

```
export PATH=$HOME/.local/bin:/cvmfs/cms-bril.cern.ch/brilconda/bin:$PATH
brilcalc lumi --normtag /cvmfs/cms-bril.cern.ch/cms-lumi-pog/Normtags/normtag_PHYSICS.json -c /c
```

## How to find unprecaled trigger:

GRL is available here:

```
/afs/cern.ch/cms/CAF/CMSCOMM/COMM_DQM/certification/Collisions17/13TeV/Final/Cert_306896-307082_1
```

To get list of runs execute: `printJSON.py`

```
/afs/cern.ch/cms/CAF/CMSCOMM/COMM_DQM/certification/Collisions17/13TeV/Final/Cert_306896-307082_1
```

To get the list of triggers per run goto <https://cmsoms.cern.ch/cms/runs>, enter a run number, and click on HLT key option.

Extra information about calculations of collision rates and pileup interactions is available in `PileupJSONFileforData`.

## Obtaining pileup rate

To be able to know the average number of collisions per bunch crossing, the most straightforward way of doing this is to use the instantaneous luminosity (see `PileupJSONFileforData` for more details).

If a single bunch has an instantaneous luminosity  $L_{inst}$ , then the pileup is given by the formula  $\mu = L_{inst} \cdot \sigma_{inel} / f_{rev}$ , where  $\sigma_{inel} = 69.2$  mb is the total pp inelastic cross section and  $f_{rev}$  is the LHC orbit frequency of 11246 Hz (necessary to convert from the instantaneous luminosity, which is a per-time quantity, to a per-collision quantity). This quantity can be computed on a per-lumi section basis (where a lumi section is the fundamental unit of CMS luminosity calculation, about 23.3 seconds long).

The average instantaneous luminosity is stored in

```
/afs/cern.ch/cms/CAF/CMSCOMM/COMM_DQM/certification/Collisions17/13TeV/PileUp/pileup_latest.txt
```

for 2017 data. To get the luminosity values you can use python script:

```
import json, numpy
with open('/afs/cern.ch/cms/CAF/CMSCOMM/COMM_DQM/certification/Collisions17/13TeV/PileUp/pileup_17.txt') as f:
    data = json.load(f)
run='306936'
lumis=numpy.array([f[3] for f in data[run] if float(f[3])>0])
print('maximal pileup = '+str(lumis.max()*69200.)+'\nminimal pileup = '+str(lumis.min()*69200.)+'')
```

## Good Run Lists:

The luminosity certified as good for physics analysis is contained in the JSON Golder files:

- 13 TeV standard runs:

```
/afs/cern.ch/cms/CAF/CMSCOMM/COMM_DQM/certification/Collisions17/13TeV/Final/Cert_294927-3
```

- 13 TeV low  $\mu$  dataset:

```
/afs/cern.ch/cms/CAF/CMSCOMM/COMM_DQM/certification/Collisions17/13TeV/Final/Cert_306896-3
```

-- MichaelPitt - 2021-07-18

This topic: Sandbox > TriggerStudy

Topic revision: r4 - 2021-07-19 - MichaelPitt



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)