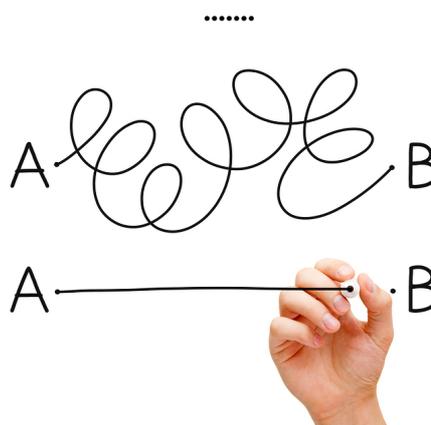


Table of Contents

What is it?.....	1
What does it add?.....	1
Disclaimer: on real data think about what you're doing first!.....	1
How to use it?.....	1
Resources.....	2
Options and settings.....	2
Input Requirements.....	2
Common use-cases:.....	3
Feature discussion.....	3

The "L0 Trigger" is a hardware and firmware based trigger, but it can be emulated to surprising accuracy in software.

(the result of task #42380 [↗](#))



What is it?

- **L0App()** is a very simple configurable which has only one use, to emulate the L0 and write out a file with L0 Added to it.
- It is able to do this to any number of MDF files, RAW files, or DSTs, and employs a combination of:
 - ◆ `InputCopyStream`
 - ◆ `L0Conf`
 - ◆ `MDFWriter`

What does it add?

- **L0App** adds or replaces L0 banks in the raw event. It will do this for any file which still has a raw event on it, such as:
 - ◆ **YES:** `DST`, `DIGI`, `XDIGI`, `XDST`, `RAW`, `MDF`
 - ◆ **NO:** `SIM`, `XSIM`, `MDST`
- On real data it behaves the same as on simulated data in that it replaces all the L0 banks with emulated banks no matter what.

Disclaimer: on real data think about what you're doing first!

When re-emulating L0 on real data, as with any trigger re-emulation consider that the data that you are running on has already been triggered. The LHCb trigger is intended to bias events towards bigger events with higher chance of finding a B-hadron for your analysis. The performance of the re-emulation on already triggered data is then biased by the output of the previous trigger and/or selection, whatever that may be. You are performing a logical AND of your two trigger conditions `Trigger1&&Trigger2`. In an extreme case, for example, re-running the same L0 on the same file should ideally not reject a single event. So, when using real data remember:

- **Either:** use truly random triggers, NoBias data
- **Or:** do not trust your naive rates, efficiencies, and performance calculations.

How to use it?

- Setup the project `SetupProject Moore`
- Set up your input
 - ◆ any gaudi-card dataset will work, in Ganga, from the book-keeping, or from the `TestFileDB...` here is an example with a file stored in the `TestFileDB`

```
#imagine this is in a file called mydata.py
from GaudiConf import IOExtension
IOExtension().inputFiles(["root://eoslhcb.cern.ch//eos/lhcb/grid/prod/lhcb/swtest/lhcb/swtest/201
```

- Use L0App

```
#imagine this is in a file called L044.py
from Configurables import L0App
L0App().TCK='0x0044'
L0App().outputFile="radiative_plus_1044.dst"
L0App().ReplaceL0Banks=True #only needed if the input file already contains L0
```

- e.g.: gaudirun.py mydata.py L044.py

Resources

- SVN package [↗](#)
- L0App configurable [↗](#)
- L0App Doxygen [↗](#)

Options and settings

- **L0App** is about the simplest possible configurable application it's possible to write in LHCb software.
- **L0App** has very few options, and all of them are very obvious...

```
class L0App(LHCbConfigurableUser):
    ## Possible used Configurables
    __used_configurables__ = [ LHCbApp
                              , L0Conf
                              , DecodeRawEvent ]

    __slots__ = {
        "EvtMax": -1 # Maximum number of events to process
        , "SkipEvents": 0
        , "Simulation": True # True implies use SimCond
        , "DataType": '2012' # Data type, can be [ '2012','2011'... ]
        , "DDDBtag" : 'default' # default as set in DDDBConf for DataType
        , "Conddbtag" : 'default' # default as set in DDDBConf for DataType
        , "ReplaceL0Banks" : False # remove existing banks first?
        , 'TCK' : '' #"Can be a L0TCK or a full TCK, it doesn't matter"
        , "outputFile" : '' # output filename
        , 'WriteFSR' : True #copy FSRs as required
    }
}
```

Input Requirements

- **Simulation:** Ideally L0App runs after digitization but before reconstruction or HLT/Moore. This is equivalent of when the real L0 runs in the pit. So, ideally supply a DIGI or XDIGI
- **Real Data:** Similarly the L0 happened before the HLT/Moore and reconstruction, and so it is best to run it on raw files whose HLT has been cleaned off and whose reconstruction has not happened.
- **Juggling:** unfortunately the raw event is split up and moved around by current Brunel and so if you're running on reconstructed data you will need to think about -
- **DAQ/RawEvent:**
 - ◆ just like Moore, L0App requires only one raw event location, and this should be DAQ/RawEvent
 - ◆ However, Brunel if run on your input file will most probably have split this up into different locations, and you need to recombine them.

- ◆ Use RawEventJuggler twiki to understand what you need to do here and why.
- ◆ **You only need to juggle files which have already been reconstructed!**
- How to tell what my input file contains?

```
from Gaudi.Configuration import *
from Configurables import LHCbApp
from Configurables import RawEventDump
from Configurables import StoreExplorerAlg

StoreExplorerAlg().Load=True
RawEventDump().DumpData=False
LHCbApp().EvtMax=10

ApplicationMgr().TopAlg+=[StoreExplorerAlg(), RawEventDump()]

#add your input files..., example here from Eric
from GaudiConf import IOExtension
IOExtension().inputFiles(['root://castorlhcb.cern.ch//castor/cern.ch/user/e/evh/bwdivision/137740
```

Common use-cases:

- Emulating L0 on fully simulated data.
- Re-emulating a new L0 on real data in order to record a TCK from a threshold setting (use NoBias data)
- Re-emulating L0 with changed thresholds to perform the bandwidth division (either NoBias data [background] or simulation [signal])
- Re-emulating L0 on real data for the purposes of testing in the nightlies (ideally use NoBias data, but this depends on the test)
- Re-emulating in order to calculate the overlap in efficiencies between two TCKs.

Feature discussion

- **Immutable:** Any L0App version should be able to emulate any/all previous L0 TCKs
 - ◆ Each night this is tested, if there is any problem emulating our firmware into the future, the L0 team need to know immediately.
- **Simple:** L0App is meant to be as simple as possible, and only to do one task.

-- RobLambert - 24 Mar 2014

This topic: LHCb > L0App

Topic revision: r6 - 2016-04-01 - FrancescaDordei



Copyright &© 2008-2019 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback