

```

#!/usr/bin/env python

#Import the necessary modules
from GaudiKernel.ProcessJobOptions import PrintOff, InstallRootLoggingHandler
import logging
InstallRootLoggingHandler(level = logging.CRITICAL)
from Gaudi.Configuration import *
import GaudiKernel.ProcessJobOptions
import Gaudi.Configuration as Gaudi
import Configurables as Configs
import OnlineEnv as Online
import os

from Configurables import (
    LHCbApp,
    ApplicationMgr,
    LHCb__MDFOnlineEvtSelector,
    LHCb__FilePoller,
    LHCb__RawDataCnvSvc,
    GaudiSequencer,
    HltMassMonitor,
    HltRoutingBitsFilter,
    HltSelReportsDecoder,
    HltDecReportsDecoder,
    HltVertexReportsDecoder,
    HltLumiSummaryDecoder,
    HltMonitor
)

app = LHCbApp()
app.Persistency="RAW"
app.EvtMax = -1

#Start ApplicationManager
appMgr = ApplicationMgr()
appMgr.EvtMax = -1
appMgr.HistogramPersistency = 'ROOT'
appMgr.SvcOptMapping.append('LHCb::FmcMessageSvc/MessageSvc')

#Set up persistency service, the service for the permanent storage of the histograms
EventPersistencySvc().CnvServices.append( LHCb__RawDataCnvSvc('RawDataCnvSvc') )
HistogramPersistencySvc().OutputFile = "
HistogramPersistencySvc().Warnings = False
ApplicationMgr().HistogramPersistency = "ROOT"

#The first file that will be produced, note that it does not have actual content
HistogramPersistencySvc().OutputFile = "testPol.root"

appMgr.TopAlg.append('StoreExplorerAlg')
StoreExplorerAlg.Load = 1
StoreExplorerAlg.PrintFreq = 100
StoreExplorerAlg.OutputLevel = 1;

```

```

data = Online.evtDataSvc()
data.RootCLID = 1
data.ForceLeaves = 1
data.EnableFaultHandler = True

#Invoke poller
poller = LHCB__FilePoller('Poller')
appMgr.ExtSvc.append(poller)

#Set the directory that the poller will scan
poller.scanDirectory = "/daqarea/lhcb/data/2014/RAW/FULL/LHCB1/TEST"

#Set the time interval between consecutive pollings
poller.alarmTime = 3

#Set the name of the database that will contain details of the files (run number etc).
poller.DbName = "./OnlineFileProcessing.db"

#Invoke EventSelector
selector = LHCB__MDFOnlineEvtSelector('EventSelector')
appMgr.ExtSvc.append(selector)

#Maximum number of events to be read from a file
selector.MaxNoEvents = 50000;

#Frequency of message printing
selector.PrintFreq = 10000

#Minimum number of events to produce a histogram
selector.EvtsForHist = 30000

# The stuff to run
physFilter = HltRoutingBitsFilter( "PhysFilter", RequireMask = [ 0x0, 0x4, 0x0 ] )
dec = HltSelReportsDecoder(SourceID=2)
vdec = HltVertexReportsDecoder()
lumidec = HltLumiSummaryDecoder()
monitor = HltMassMonitor()

monitor.Decisions = { "Jpsi"      : "Hlt2DiMuonJPsiDecision",
                     "D0->Kpi"   : "Hlt2CharmHadD02HH_D02KPiDecision",
                     "D0->KK"    : "Hlt2CharmHadD02HH_D02KKDecision",
                     "D0->pipi"   : "Hlt2CharmHadD02HH_D02PiPiDecision",
                     "D->hhh"     : "Hlt2CharmHadD2HHHDecision",
                     "D->hhhh"    : "Hlt2CharmHadD02HHHHDDecision",
                     "InclusivePhi" : "Hlt2IncPhiDecision" }
monitor.Histograms = { "Jpsi"      : [ 3005, 3051, 3141, 3186, 50 ],
                     "D0->Kpi"   : [ 1815, 1840, 1890, 1915, 50 ],
                     "D0->KK"    : [ 1815, 1840, 1890, 1915, 50 ],
                     "D0->pipi"   : [ 1815, 1840, 1890, 1915, 50 ],
                     "D->hhh"     : [ 1815, 1840, 1890, 1915, 50 ],
                     "D->hhhh"    : [ 1815, 1840, 1890, 1915, 50 ],

```

```
"InclusivePhi" : [ 990, 1005, 1035, 1050, 30 ] }
```

```
# Top level sequence, the main algorithm needed to run the monitoring programs
```

```
topSeq = GaudiSequencer( "TopSequence" )
```

```
topSeq.Members = [ dec, vdec, lumidec, monitor ]
```

```
appMgr.TopAlg = [ topSeq ]
```

```
#Define the level of detail of the messages, 0 (no messages) - 7 (always print message)
```

```
appMgr.OutputLevel = 3;
```