

Table of Contents

L1 Trigger Data Quality Monitoring (L1 DQM).....	1
Introduction.....	1
L1 DQM Structure.....	1
Concise Description of L1 DQM Code Packages.....	1
DQM/L1TMonitor.....	1
DQM/L1TMonitorClient.....	1
Contacts.....	2
L1 DQM Source Contacts.....	2
TPG DQM Contacts (Sources included in Detector DQM).....	2
Shifter Instructions for Online L1T and L1TEMU DQM.....	2
Online DQM Gui servers for L1T and L1TEMU.....	2
Official configuration files for online L1T and L1TEMU.....	2
Shifter Instructions.....	2
L1T Layout description.....	3
L1TEMU Layout description.....	3
DQM Sources Description.....	4
GT monitor.....	4
FED monitor.....	4
Timing monitor.....	4
RCT monitor.....	5
GCT monitor.....	5
GMT monitor.....	6
DT TF monitor.....	7
CSC TF monitor.....	9
RPC TF monitor.....	9
Offline L1 DQM Documentation.....	10
Technical details for L1 Trigger DQM developers.....	10
Implementing Quality Tests.....	10
Testing L1 Trigger emulator DQM () and L1 Trigger DQM () based on quality tests.....	11
Review status.....	12

L1 Trigger Data Quality Monitoring (L1 DQM)

Introduction

L1 DQM Structure

The L1 Trigger Data Quality Monitoring has more components:

- monitor the L1 trigger systems via histograms produced on L1 trigger data taken (workspace L1T, folder L1T)
- compare the L1 trigger data taken with the results of the L1 trigger emulator (workspace L1TEMU, folder L1TEMU)
- compare the L1 trigger data taken with reconstructed objects (folder L1TvsReco in L1T workspace), including efficiency calculation - under development
- compare the L1 trigger data taken with generated objects (folder L1TvsGen in L1T workspace), including efficiency calculation - under development
- compare, for the objects used in L1TvsReco and L1TvsGen, the reconstructed objects with generated objects (folder L1TRecoVsGen in L1T workspace) - under development

L1 DQM workflows run:

- online at P5, during data taking: L1T and L1TEMU
- offline at Tier-0: L1T, L1TEMU, L1TvsReco (to be updated)
- offline, re-reco: TBD
- offline, for release validation samples: L1T, L1TEMU (data RelVal), L1TvsGen (MC RelVal), L1TvsReco, L1TRecoVsGen(MC RelVal)
- offline, for MC production: L1T, L1TvsGen, L1TvsReco, L1TRecoVsGen(MC RelVal)

From the code point of view, one has the following packages

- DQM/L1TMonitor
- DQM/L1TMonitorClient
- DQMOffline/L1Trigger
- Validation/L1Trigger

Concise Description of L1 DQM Code Packages

DQM/L1TMonitor

The L1TMonitor package collects all the DQM sources, that is, all the edm analyzers which produce the Monitor Elements (ME) from the different L1 Trigger subsystems. In the current schema the MEs are then shipped to a Collector which is connected to the central DQM GUI. This allows the shifters (or the experts) to access all the information that is produced by the sources.

Follow the CVS link [↗](#) for the list of the source code of the available DQM sources.

DQM/L1TMonitorClient

The L1TMonitorClient package contains the DQM clients for the L1 Trigger systems. The client can access all the MEs produced by the sources and perform Quality Tests (QT) on them. Also, the Client is an edm Analyzer and therefore can produce MEs as well as access DB information via EventSetup. On the basis of the QT result the status of a given ME is decided. Warnings or errors can be displayed by the central GUI with a defined color coding. The results of a QT are attached to the ME.

The source code can be found via the CVS link [↗](#).

Contacts

L1 DQM Source Contacts

Subsystem	Contact
Online DQM	VasileGhete, PasqualeMusella
Offline DQM	VasileGhete, PasqualeMusella
DTF	Carlo Batillina, Jorge Troconiz
CSCTF	Joe Gartner, Gian Piero Di Giovanni
RPC	Tomasz Fruboes
RCT	Sascha Savin
GCT	Alex Tapper
GM	Ivan Mikulec
GT	VasileGhete
BxTiming	Andre Holzner
Emulator	VasileGhete

TPG DQM Contacts (Sources included in Detector DQM)

Subsystem	Contact
ECAL TPG	Pascal Paganini
HCAL TPG	Tulika Bose, Jane Nachtman, Terrence Toole
DT TPG	Stefano Marcellini
CSC TPG	Andrey Korytov, Victor Barashko

Shifter Instructions for Online L1T and L1TEMU DQM

Online DQM Gui servers for L1T and L1TEMU

The list of DQM servers is maintained by the central DQM group. They are summarized DQM P5 Installation

Official configuration files for online L1T and L1TEMU

The "official" configuration files are in DQM/Integration/test (please don't modify these files without the explicit accept of the L1 DQM coordinator(s) or by L1 Trigger offline coordinators(s)). The version used is tagged as `ONLINE_PROD`

- L1T: [l1t_dqm_sourceclient-live_cfg.py](#) [↗](#)
- L1TEMU: [l1temulator_dqm_sourceclient-live_cfg.py](#) [↗](#)

Shifter Instructions

Here you can find a description of the most important histograms that the shifter should look at. A collection of these histograms is displayed in the L1 Trigger layouts which are displayed by the DQM GUI. Each trigger subsystem should implement one or two of these histograms which should be sensitive to problems and easy

to understand (ex. occupancy plots). Also each histogram of should have an automated Quality Test associated to it.

L1T Layout description

- GT: **Regional_trigger**, **algo_bits**
- GMT: **GMT_eta**, **GMT_phi**
- CSC: **CSCTF_errors** - monitors CSC TF data integrity, error bits for our input LCTs (synchronization related), and FMM status. For simplicity, all errors are logically ORed; **CSCTF_occupancies** - scatter plot in eta x kind-a-phi plane for input LCTs and output tracks.
- RPCTF: not yet provided.
- DTF:
 - ◆ 2ndTrack_Summary: number of 2nd tracks (proportionate to total number of events that have a DTF track) - Sector vs Logical Wheel
 - ◆ bx_Summary: summary of BX's for each wheel (proportionate to total number of events with a DTF track)
 - ◆ highQual_Summary: fractional high quality (qual>3) (proportional to total bin occupancy) summary - Sector vs Logical Wheel
 - ◆ dtf_bx_integ: integrated DTF BX
 - ◆ dtf_p_q_integ: integrated packed charge
 - ◆ dtf_p_eta_integ: integrated packed eta
 - ◆ dtf_p_phi_integ: integrated packed phi
 - ◆ dtf_p_pt_integ: integrated packed pt
 - ◆ dtf_p_qual_integ: integrated packed quality
 - ◆ dtf_nTracksPerEvent_integ: integrated number of tracks per event
 - ◆ dtf_p_phi_eta: integrated occupancy (proportionate to total number of events with a DTF track) - phi vs eta
 - ◆ Occupancy_Summary: integrated occupancy (proportionate to total number of events with a DTF track) - Sector vs Logical Wheel
- GCT: : eta-phi occupancy histograms, weighted by rank(=et): **ISO EM RANK** = isolated electrons passing through the GCT for ECAL data; **NON-ISO EM RANK** = non-isolated electrons for HCAL data.
- RCT: not yet provided.
- ECAL:
- HCAL:

L1TEMU Layout description

The emulator DQM is based on standard data-emulation comparison. A common set of monitoring elements (histograms) is correspondingly generated for the various subsystems (which include trigger primitives, regional and global trigger stages). The 'layout' set is defined to contain one five-bin histogram, per subsystem, conveying whether there was agreement or otherwise classifying the type of disagreement. The bin labels and definitions are as follows:

- *agree* (full agreement),
- *loc. agree* (eta, phi, .. position coordinates agree),
- *loc. disagree* (position does not agree),
- *data only* (trigger objects found in data but missing in emulation),
- *emul only* (trigger objects not found in data but predicted by emulation).

These histograms are named **ErrorFlag** prefixed with a subsystem identifier, ETP, HTP, RCT, GCT, DTP, DTF, CTP, CTF, RPC, LTC, GMT, GLT while further information (eg eta-phi map of disagreements, un-matched trigger-data bits) is available under the subsystem folders, correspondingly named ECAL, HCAL, RCT, GCT, DTPG, DTF, CSCTPG, CSCTF, RPC, LTC, GMT, GT. A quality test checks for each case

wether the fraction of entries in the agreement bin is above threshold.

DQM Sources Description

Central support: VasileGhete, PasqualeMusella

GT monitor

Module names: L1TGT

- Source code: [DQM/L1TMonitor/src/L1TGT.cc](#)
- Client code:
- Data products monitored: L1CMS.GlobalTriggerReadoutRecord (L1GtfeWord, L1GtFdlWord, DecisionWord)
- Monitorables:
 - ◆ GT decision word: full vector of decision bits as a 128-bit integer (TH1F)
 - ◆ GT decision bits: bit-wise projection of decision word (TH1F)
 - ◆ GT decision bit correlation: pairwise correlation of bits (TH2F)
 - ◆ GT FDL Bx: bx number for FDL readout
 - ◆ GT FDL Event: event number for FDL readout
 - ◆ GT FDL local Bx: local bx number for fdl readout
 - ◆ GT FDL Bxinevent: bxinevent for FDL readout
 - ◆ GT FDL size: size of FDL readout
 - ◆ GT FE board Id: FE board ID number
 - ◆ GT FE record length: FE record length
 - ◆ GT FE Bx: bx number of FE readout
 - ◆ GT FE setup version: version number of FE firmware
 - ◆ GT FE active boards: number of active boards for FE readout
 - ◆ GT FE total triggers: trigger count for FE readout
 - ◆ GT FE size: size of FE readout

FED monitor

Module names: L1TFED

- Source code: [DQM/L1TMonitor/src/L1TFED.cc](#)
- Client code:
- Data products monitored: FEDRawDataCollection, FEDRawData
- Monitorables:
 - ◆ fedprof: FED size (TH1F)
 - ◆ fedsizes: FED size vs. FED number (Profile)
 - ◆ Details/fedxxx: FED size for FED #xxx

Timing monitor

Module names: BxTiming

- Source code: [DQM/L1TMonitor/interface/BxTiming.h](#)
- Client code:
- Data products monitored: FEDRawDataCollection, L1CMS.GlobalTriggerReadoutRecord
- Monitorables:
 - ◆ BxDiffAllFed: bx shift wrt reference fed, for all feds
 - ◆ BxDiffSysFed[]: bx shift wrt reference fed, per subsystem
 - ◆ BxDiffAllFedSpread[]: bx shift wrt ref fed: mean shift, min, max
 - ◆ BxOccyAllFed: bx occupancy, for all fed's

- ◆ BxOccyAllFedSpread[]: bx occupancy: mean shift, min, max
- ◆ BxOccyOneFed: bx occupancy, per each fed
- ◆ BxOccyGtTrigType[]: gt bx occupancy per trigger type
- ◆ BxOccyTrigBit[][]: subsystem bx occupancy per selected trigger bit
- Run/test: cfi [↗](#) cfg [↗](#) display [↗](#)

RCT monitor

Module names: L1TRCT

- Source code: [DQM/L1TMonitor/src/L1TRCT.cc](#) [↗](#)
- Client code:
- Data products monitored: L1CaloEmCollection, L1CaloRegionCollection
- Monitorables:
 - ◆ RctIsoEmEtEtaPhi: Isolated EM cluster rank-weighted occupancy
 - ◆ RctIsoEmOccEtaPhi: Isolated EM cluster occupancy
 - ◆ RctNonIsoEmEtEtaPhi: Non-isolated EM cluster rank-weighted occupancy
 - ◆ RctNonIsoEmOccEtaPhi: Non-isolated EM cluster occupancy
 - ◆ RctRegionsEtEtaPhi: Calo regions rank-weighted occupancy
 - ◆ RctRegionsOccEtaPhi: Calo regions occupancy
 - ◆ RctTauVetoEtaPhi: Tau veto region occupancy
 - ◆ RctRegionsLocalEtEtaPhi: Calo regions rank-weighted occupancy (in local coords)
 - ◆ RctRegionsLocalOccEtaPhi: Calo regions occupancy (in local coords)
 - ◆ RctTauLocalVetoEtaPhi: Tau veto region occupancy (in local coords)
 - ◆ RctRegionRank: Calo region rank distribution
 - ◆ RctIsoEmRank: Isolated EM cluster rank
 - ◆ RctNonIsoEmRank: Non-isolated EM cluster rank
 - ◆ rctEmCardRegion: Em card*region
 - ◆ RctRegionBx: bx number distributions of all calo regions
 - ◆ RctEmBx: bx number distribution of all EM clusters

GCT monitor

Module names: L1TGCT

- Source code: [DQM/L1TMonitor/src/L1TGCT.cc](#) [↗](#)
- Client code: [DQM/L1TMonitorClient/src/L1TGCTClient.cc](#) [↗](#)
- Data products monitored: L1GctEmCandCollection, L1GctEmCandCollection, L1GctJetCandCollection, L1GctJetCandCollection, L1GctJetCandCollection, L1GctEtMiss, L1GctEtHad, L1GctEtTotal
- Monitorables:
 - ◆ CenJetsEtEtaPhi: central jets rank-weighted occupancy
 - ◆ CenJetsOccEtaPhi: central jets occupancy
 - ◆ CenJetsRank: central jets rank
 - ◆ EtHad: total HAD ET
 - ◆ EtMiss: missing ET
 - ◆ EtMissPhi: missing ET phi
 - ◆ EtTotal: sum ET
 - ◆ ForJetsEtEtaPhi: forward jets rank-weighted occupancy
 - ◆ ForJetsOccEtaPhi: forward jets occupancy
 - ◆ ForJetsRank: forward jets rank
 - ◆ GctIsoEmRankCand0: rank of top-ranked isolated EM cluster
 - ◆ GctIsoEmRankCand1: rank of second-ranked isolated EM cluster
 - ◆ GctIsoEmRankCand2: rank of third-ranked isolated EM cluster
 - ◆ GctIsoEmRankCand3: rank of fourth-ranked isolated EM cluster

- ◆ GctIsoEmRankDiffCand01: rank difference of isolated EM clusters 0 and 1
- ◆ GctIsoEmRankDiffCand12: rank difference of isolated EM clusters 1 and 2
- ◆ GctIsoEmRankDiffCand23: rank difference of isolated EM clusters 2 and 3
- ◆ GctNonIsoEmRankCand0: rank of top-ranked non-isolated EM cluster
- ◆ GctNonIsoEmRankCand1: rank of second-ranked non-isolated EM cluster
- ◆ GctNonIsoEmRankCand2: rank of third-ranked non-isolated EM cluster
- ◆ GctNonIsoEmRankCand3: rank of fourth-ranked non-isolated EM cluster
- ◆ GctNonIsoEmRankDiffCand01: rank difference of isolated EM clusters 0 and 1
- ◆ GctNonIsoEmRankDiffCand12: rank difference of isolated EM clusters 1 and 2
- ◆ GctNonIsoEmRankDiffCand23: rank difference of isolated EM clusters 2 and 3
- ◆ IsoEmOccEtaPhi: isolated EM cluster occupancy
- ◆ IsoEmRank: isolated EM cluster rank
- ◆ IsoEmRankEtaPhi: isolated EM cluster rank-weighted occupancy
- ◆ NonIsoEmOccEtaPhi: non-isolated EM cluster occupancy
- ◆ NonIsoEmRank: non-isolated EM cluster rank
- ◆ NonIsoEmRankEtaPhi: non-isolated EM cluster rank-weighted occupancy
- ◆ TauJetsEtEtaPhi: tau jet rank-weighted occupancy
- ◆ TauJetsOccEtaPhi: tau jet occupancy
- ◆ TauJetsRank: tau jet rank
- ◆ ISO EM/GctIsoEmRank-x-y: isolated EM cluster rank in eta=x, phi=y
- ◆ NON-ISO EM/GctNonIsoEmRank-x-y: non-isolated EM cluster rank in eta=x, phi=y

GMT monitor

Module names: L1TGMT

- Source code: [DQM/L1TMonitor/src/L1TGMT.cc](#)
- Client code: [DQM/L1TMonitorClient/src/L1TGMTClient.cc](#)
- Data products monitored: L1MuGMTReadoutCollection (for some number of bx, the input L1MuRegionalCands from the track-finders and the L1MuGMTExtendedCands output by the GMT)
- Monitorables:
 - ◆ CSCTF_bits: CSCTF bit population
 - ◆ CSCTF_dbx: dBX CSCTF to previous event
 - ◆ CSCTF_eta: CSCTF eta value
 - ◆ CSCTF_etaphi: CSCTF phi vs eta
 - ◆ CSCTF_nbx: CSCTF multiplicity in bx
 - ◆ CSCTF_phi: CSCTF phi value
 - ◆ CSCTF_ptr: CSCTF pt value
 - ◆ CSCTF_qty: CSCTF quality
 - ◆ DTFE_bits: DTFE bit population
 - ◆ DTFE_dbx: dBX DTFE to previous event
 - ◆ DTFE_eta: DTFE eta value
 - ◆ DTFE_etaphi: DTFE phi vs eta
 - ◆ DTFE_nbx: DTFE multiplicity in bx
 - ◆ DTFE_phi: DTFE phi value
 - ◆ DTFE_ptr: DTFE pt value
 - ◆ DTFE_qty: DTFE quality
 - ◆ GMT_bits: GMT bit population
 - ◆ GMT_eta: GMT eta value
 - ◆ GMT_etaphi: GMT phi vs eta
 - ◆ GMT_nbx: GMT multiplicity in bx
 - ◆ GMT_phi: GMT phi value
 - ◆ GMT_ptr: GMT pt value
 - ◆ GMT_qty: GMT quality
 - ◆ RPCb_bits: RPCb bit population

- ◆ RPCb_dbx: dBX RPCb to previous event
- ◆ RPCb_eta: RPCb eta value
- ◆ RPCb_etaphi: RPCb phi vs eta
- ◆ RPCb_nbx: RPCb multiplicity in bx
- ◆ RPCb_phi: RPCb phi value
- ◆ RPCb_ptr: RPCb pt value
- ◆ RPCb_qty: RPCb quality
- ◆ RPCf_bits: RPCf bit population
- ◆ RPCf_dbx: dBX RPCf to previous event
- ◆ RPCf_eta: RPCf eta value
- ◆ RPCf_etaphi: RPCf phi vs eta
- ◆ RPCf_nbx: RPCf multiplicity in bx
- ◆ RPCf_phi: RPCf phi value
- ◆ RPCf_ptr: RPCf pt value
- ◆ RPCf_qty: RPCf quality
- ◆ n_CSCTF_vs_DTTF: n candS CSCTF vs DTTF
- ◆ n_RPCb_vs_DTTF: n candS RPCb vs DTTF
- ◆ n_RPCf_vs_CSCTF: n candS RPCf vs CSCTF

DT TF monitor

Module names: L1TDTF

- Source code: [DQM/L1TMonitor/src/L1TDTF.cc](#)
- Client code:
- Data products monitored: L1MuDTTrackContainer
- Monitorables:
 - ◆ INTEG
 - ◇ 2ndTrack_Summary: number of 2nd tracks (proportionate to total number of events that have a DTTF track) - Sector vs Logical Wheel
 - ◇ bx_Summary: summary of BX's for each wheel (proportionate to total number of events with a DTTF track)
 - ◇ highQual_Summary: fractional high quality (qual>3) (proportional to total bin occupancy) summary - Sector vs Logical Wheel
 - ◇ dttf_bx_integ: integrated DTTF BX
 - ◇ dttf_p_q_integ: integrated packed charge
 - ◇ dttf_p_eta_integ: integrated packed eta
 - ◇ dttf_p_phi_integ: integrated packed phi
 - ◇ dttf_p_pt_integ: integrated packed pt
 - ◇ dttf_p_qual_integ: integrated packed quality
 - ◇ dttf_nTracksPerEvent_integ: integrated number of tracks per event
 - ◇ dttf_p_phi_eta: integrated occupancy (proportionate to total number of events with a DTTF track) - phi vs eta
 - ◇ Occupancy_Summary: integrated occupancy (proportionate to total number of events with a DTTF track) - Sector vs Logical Wheel
 - ◇ Occupancy_Summary_r: same as Occupancy_Summary, but resets every monitoring cycle
 - ◇ 2ND_TRACK_ONLY
 - bx_Summary_2ndTrack: proportional to total number of events that have a 2nd DTTF track
 - highQual_Summary_2ndTrack: proportional to 2nd track bin occupancy
 - dttf_bx_integ_2ndTrack
 - dttf_nTracks_integ_2ndTrack
 - dttf_p_q_integ_2ndTrack
 - dttf_p_eta_integ_2ndTrack

- dttf_p_phi_integ_2ndTrack
- dttf_p_pt_integ_2ndTrack
- dttf_p_qual_integ_2ndTrack
- dttf_p_phi_eta_2ndTrack: proportional to total number of events with a 2nd DTF track
- Occupancy_Summary_2ndTrack: proportional to total number of events with a 2nd DTF track
- ◆ WHEEL_N2
 - ◇ dttf_n2ndTracks_whN2
 - ◇ dttf_bx_whN2
 - ◇ dttf_nTracksPerEvent_whN2
 - ◇ dttf_nTracks_whN2
 - ◇ dttf_p_phi_eta_whN2
 - ◇ BX_0
 - Charge
 - dttf_p_q_bx0_whN2_se1
 - ...
 - Eta
 - dttf_p_eta_bx0_whN2_se1
 - ...
 - PT
 - dttf_p_pt_bx0_whN2_se1
 - ...
 - Phi
 - dttf_p_phi_bx0_whN2_se1
 - ...
 - Quality
 - dttf_p_qual_bx0_whN2_se1
 - ...
 - TracksPerEvent
 - dttf_p_nTrksPerEv_bx0_whN2_se1
 - ...
 - ◇ BX_N1
 - ...
 - ◇ BX_P1
 - ...
 - ◇ BX_SECTORS
 - ALL
 - dttf_bx_whN2_se1
 - ...
 - TRACK_2_ONLY
 - dttf_bx_2ndTrack_whN2_se1
 - ...
- ◆ WHEEL_N1
 - ◇ ...
- ◆ WHEEL_N0
 - ◇ ...
- ◆ WHEEL_P0
 - ◇ ...
- ◆ WHEEL_P1
 - ◇ ...
- ◆ WHEEL_P2
 - ◇ ...

CSC TF monitor

Module names: L1TCSTF

- Source code: [DQM/L1TMonitor/src/L1TCSTF.cc](#)
- Client code: [DQM/L1TMonitorClient/src/L1TCSTFClient.cc](#)
- Data products monitored: L1MuRegionalCands from L1MuGMTReadoutCollection
- Monitorables:
 - ◆ CSCTF_bx: CSCTF bx
 - ◆ CSCTF_charge_value: CSCTF charge value
 - ◆ CSCTF_charge_value_+1: CSCTF charge value bx +1
 - ◆ CSCTF_charge_value_-1: CSCTF charge value bx -1
 - ◆ CSCTF_eta_value: CSCTF eta value
 - ◆ CSCTF_eta_value_+1: CSCTF eta value bx +1
 - ◆ CSCTF_eta_value_-1: CSCTF eta value bx -1
 - ◆ CSCTF_ntrack: CSCTF ntrack
 - ◆ CSCTF_phi_value: CSCTF phi value
 - ◆ CSCTF_phi_value_+1: CSCTF phi value bx +1
 - ◆ CSCTF_phi_value_-1: CSCTF phi value bx -1
 - ◆ CSCTF_pt_value: CSCTF pt value
 - ◆ CSCTF_pt_value_+1: CSCTF pt value bx +1
 - ◆ CSCTF_pt_value_-1: CSCTF pt value bx -1
 - ◆ CSCTF_quality: CSCTF quality
 - ◆ CSCTF_quality_+1: CSCTF quality bx +1
 - ◆ CSCTF_quality_-1: CSCTF quality bx -1

Detailed description of the plots is available at
<https://twiki.cern.ch/twiki/bin/view/CMS/CSCTFDataQualityMonitoring>

RPC TF monitor

Module names: L1TRPCTF

- Source code: [DQM/L1TMonitor/src/L1TRPCTF.cc](#)
- Client code: [DQM/L1TMonitorClient/src/L1TRPCTFClient.cc](#)
- Data products monitored: L1MuRegionalCands from L1MuGMTReadoutCollection
- Monitorables:
 - ◆ **NOTE:** most histograms in RPCTF come in three flavours - for muon candidates with bx=0 ("*_bx0" histograms), for candidates with bx>0 ("*_bx+" histograms) and for candidates with bx<0 ("*_bx-" histograms). For better readability in the list below only histograms with bx=0 ("*_bx0") are mentioned
 - ◆ **CrateSynchroHistograms** - directory containing 12 histograms (one per trigger crate) showing crate synchronization.
 - ◆ **RPCTF_bx:** bx distribution of RPC muon candidates
 - ◆ **RPCTF_bx_diff:** bx difference between consecutive events in which RPC muon occurred
 - ◆ **RPCTF_charge_value_bx0:** charge value of RPC muon candidates
 - ◆ **RPCTF_eta_value_bx0:** eta value of RPC muon candidates
 - ◆ **RPCTF_muons_eta_phi_bx0:** eta - phi distribution of muon RPC candidates
 - ◆ **RPCTF_ntrack_brl_bx0:** number of RPC muons for barrel
 - ◆ **RPCTF_ntrack_fwd_bx0:** number of RPC muons for endcaps
 - ◆ **RPCTF_phi_value_bx0:** phi value
 - ◆ **RPCTF_pt_value_bx0:** pt value distribution of muon RPC candidates
 - ◆ **RPCTF_quality_bx0:** quality distribution of muon RPC candidates
 - ◆ **RPCTF_quality_vs_eta_bx0:** quality distribution vs eta of muon RPC candidates
 - ◆ **RPCTF_rate_avg:** average rate coming from RPC trigger (time in minutes from start of run)

- ◆ **RPCTF_rate_max**: maximal rate coming from RPC trigger during one minute periods
- ◆ **RPCTF_rate_min**: minimal rate coming from RPC trigger during one minute periods

Offline L1 DQM Documentation

The documentation for the offline L1 DQM workflows L1TvsGen, L1TvsReco, L1TRecoVsGen can be found in the following wiki pages

- L1TvsReco: L1 trigger data versus reconstructed objects
- L1TvsGen: L1 trigger data versus generated objects
- L1TRecoVsGen: reconstructed objects versus generated objects for relevant L1TvsReco and L1TvsGen objects

Technical details for L1 Trigger DQM developers

Implementing Quality Tests

The DQM framework provides a list of quality tests that can be performed on a given histogram (or MonitorElement). In order to run a QT, an XML file is needed, defining the tests and the histogram(s) to which these tests are applied. For a given QT type, like for example *MeanWithinExpected*, you can define your specific tests by choosing a set of parameters and defining a name for this test:

```
<QTEST name="myTest">
  <TYPE>MeanWithinExpected</TYPE>
  <PARAM name="error">0.05</PARAM>
  <PARAM name="warning">0.5</PARAM>
  <PARAM name="mean">0.0</PARAM>
  <PARAM name="useRMS">0</PARAM>
  <PARAM name="useSigma">0</PARAM>
  <PARAM name="useRange">1</PARAM>
  <PARAM name="xmin">-0.015</PARAM>
  <PARAM name="xmax">0.015</PARAM>
</QTEST>
```

You must then associate to histograms the quality tests (via the test name) which will be performed on those histograms. For example if you want to perform *myTest* on all the histograms containing the string *histoXYZ* you do the following:

```
<LINK name="*histoXYZ*">
  <TestName activate="true">myTest</TestName>
</LINK>
```

The QualityTester analyzers performing the quality tests on a system are to be found in DQM/L1TMonitorClient/python. There is normally no need to make any change in these files. They are collected in a single sequence **by the L1 trigger DQM coordinators** for:

- L1T: L1TriggerQualityTests_cff.py
- L1TEMU: L1EmulatorQualityTests_cff.py

The quality tests for all L1 trigger systems, L1 trigger objects and any other L1 trigger monitoring module are defined in DQM/L1TMonitorClient/data. Please define in your XML file all the quality tests needed.

File names for L1T

- quality tests analyzer (CVS link [DQM/L1TMonitorClient/python](#)):
 - ◆ L1TriggerSystem name/]QualityTests_cfi.py
 - ◆ L1TriggerObj[Trigger object]QualityTests_cfi.py
 - ◆ L1Trigger[any other L1 trigger monitoring module]QualityTests_cfi.py
- quality test definition - **here one must define the needed quality tests** - CVS link: [DQM/L1TMonitorClient/data](#)
 - ◆ L1Trigger[System name]QualityTests.xml
 - ◆ L1TriggerObj[Trigger object]QualityTests.xml
 - ◆ L1Trigger[any other L1 trigger monitoring module]QualityTests.xml

File names for L1TEMU

- quality tests analyzer (CVS link [DQM/L1TMonitorClient/python](#)):
 - ◆ L1Emulator[System name/]QualityTests_cfi.py
 - ◆ L1EmulatorObj[Trigger object]QualityTests_cfi.py
 - ◆ L1Emulator[any other L1 trigger monitoring module]QualityTests_cfi.py
- quality test definition - **here one must define the needed quality tests** - CVS link: [DQM/L1TMonitorClient/data](#)
 - ◆ L1Emulator[System name]QualityTests.xml
 - ◆ L1EmulatorObj[Trigger object]QualityTests.xml
 - ◆ L1Emulator[any other L1 trigger monitoring module]QualityTests.xml

Testing L1 Trigger emulator DQM () and L1 Trigger DQM () based on quality tests

The recipe to test the redesigned L1 Trigger DQM based on quality tests (see V.M. Ghete talk) is given below. The scripts will be maintained in the given location. The "master page" for testing is DQM documentation, where detailed information to test the tags is given - the scripts given here try to simplify your work.

Please replace in the following `my_CMSSW_RELEASE` with the actual release used at P5 for online DQM, see DQM P5 Tag Collector. The latest release I tested is `CMSSW_5_2_3_patch1`

Scripts are to be run under `zsh` shell.

1. Install CMSSW release and additional tags used at P5 and run L1TEMU on some lxplus machine
 - `YOUR_DIR` is the directory where you install CMSSW
 - 1. Set the variables containing the directory where the scripts are located and the CMSSW release you want to use

1. On lxplus

```
dqmSetupDir=/afs/cern.ch/user/g/ghete/public/L1Trigger/DQM;
my_CMSSW_RELEASE=CMSSW_5_2_3_patch1
```

2. On the online cluster (.cms at P5) get **first** a kerberos ticket in CERN realm like (note that CERN.CH is in upper case):

```
kinit username@CERN.CH
```

then proceed with the scripts

```
dqmSetupDir=/nfshome0/ghete/L1Trigger/DQM;
my_CMSSW_RELEASE=CMSSW_5_2_3_patch1
```

2. Install the release and the additional tags running at P5, plus the new L1 trigger tags

```
source ${dqmSetupDir}/scripts/install_CMSSW_P5DQM.sh ${my_CMSSW_RELEASE}
```

Before installing the release, check that **P5TagsDQM.sh** is linked to a correct **P5TagsDQM_CMSSW_RELEASE_RunXXXXXX.sh** file.

```
P5TagsDQM.sh -> P5Tags_CMSSW_5_2_3_patch1_RunXYZZZZ.sh
```

Check also that the global tag in `l1temulator_dqm_sourceclient-file_cfg.py` and / or `l1t_dqm_sourceclient-file_cfg.py` corresponds to a correct global tag for the release you are using.

3. Run the clients

1. Run `l1temulator_dqm_sourceclient-file_cfg.py` or `l1t_dqm_sourceclient-file_cfg.py`

```
source ${dqmSetupDir}/scripts/Run_L1TDQM_CMSSW_P5DQM.sh ${my_CMSSW_R
or
source ${dqmSetupDir}/scripts/Run_L1TDQM_CMSSW_P5DQM.sh ${my_CMSSW_R
```

If you run with argument "file", you will get a root file (depending on the input)

```
/YOUR_DIR/CMSSW_RELEASE/src/DQM/L1TMonitor/test/DQM_V0001_L1TEMU_R00
```

2. Run the L1T live client `l1t_dqm_sourceclient-live_cfg.py` and the L1TEMU live client `l1temulator_dqm_sourceclient-live_cfg.py` following the instructions from Setting up the online source application to feed from the playback server.

```
source ${dqmSetupDir}/scripts/Run_L1TDQM_CMSSW_P5DQM.sh ${my_CMSSW_R
or
source ${dqmSetupDir}/scripts/Run_L1TDQM_CMSSW_P5DQM.sh ${my_CMSSW_R
```

2. Render plugin testing

Work on a specific `lxplusXYZ` node in `myDir=/tmp/$USER` Use instructions from Local instance of GUI server, layouts & render plugins

3. Various changes

1. If you want to change something in the render plugin
 - ◇ make the changes
 - ◇ restart the GUI (it will compile the new render plugins)

```
source ${dqmSetupDir}/scripts/restart_GUI.sh
```

2. If you want to change the sample to run on or the number of events, edit `CMSSW/DQM/L1TMonitor/python/inputsource_file_cfi.py` Then follow the instructions from the Local instance of GUI server, layouts & render plugins.

Review status

Reviewer/Editor and Date	Comments
MVA(Created) - 10 Oct 2007	created page
LorenzoAgostino	Multiple revisions 2007 - 2008
JeffreyBerryhill	Multiple revisions 2008
NunoLeonardo	Multiple revisions 2008
MonicaVazquezAcosta	Multiple revisions 2008
VasileGhete - 24-Nov-2010	Added L1TEMU QT recipe
VasileGhete	Multiple revisions 2010 -

Responsible: VasileGhete

This topic: CMSPublic > SWGuideL1DQM

Topic revision: r82 - 2012-04-12 - VasileGhete



Copyright &© 2008-2020 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](#)