

EMT Monitoring and Analysis Software

PART 1 : Offline fundamentals

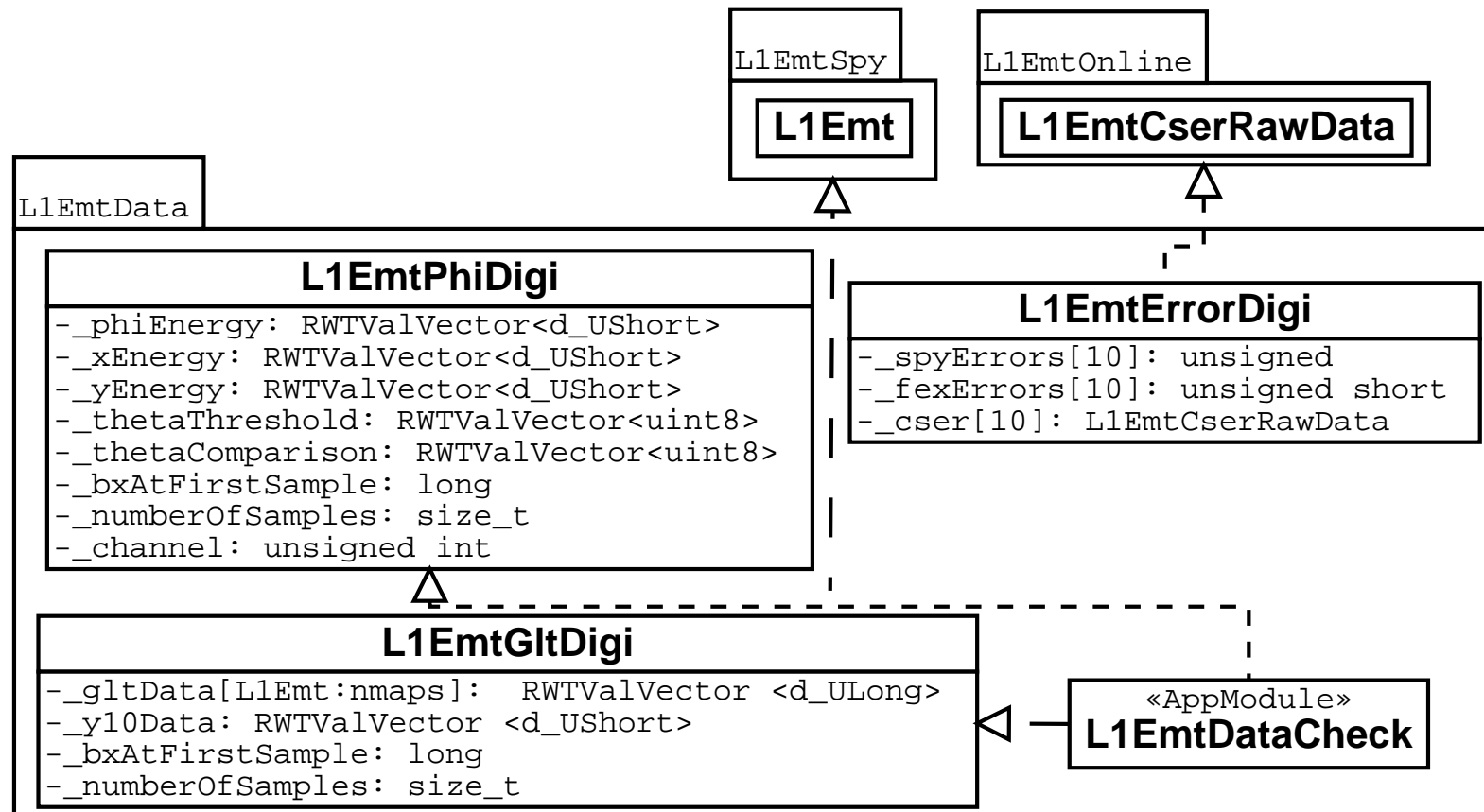
EMT SOFTWARE WORKSHOP

18th November 2001

Wahid Bhimji

L1EmtData

Digis: Offline containers for EMT information.



L1EmtData : L1EmtGltDigi

Contains information the EMT sent to the GLT.
(ie. the 20 fold Global Maps in M,G,E,X and 10-fold in Y.) 1 per Event

Contains:

- `_gltData`: Array of 5 20-bit maps for each of 32 timesamples.
- `_y10Data`: written by `sety10data()` run in constructor which does bitwise OR.
- `_bxAtFirstSample`: `clk60AtFirstGltSample` from SlotTC
- `_numberOfSamples`: 32 (hardcoded in `L1EmtTCToDigi`)

L1EmtData : L1EmtPhiDigi

40 per Event.

Mostly information sent out on DAQ stream.

Contains:

- `_phiEnergy`: Vector 16 timesamples long;
- `_xEnergy`.
- `_yEnergy`.
- `_thetaThreshold`;
- `_thetaComparison`;
- `_bxAtFirstSample` : `clk60AtFirstChannelSample` from SlotTC
- `_numberOfSamples` : 16 (hardcoded in `L1EmtTCToDigi`)

L1EmtData

L1EmtErrorDigi

1 Per Event. Contains:

- `_spyErrors[10];`
- `_fexErrors[10];`
- `_cser[10];`

Future: Replace L1EmtCserRawData with just a:

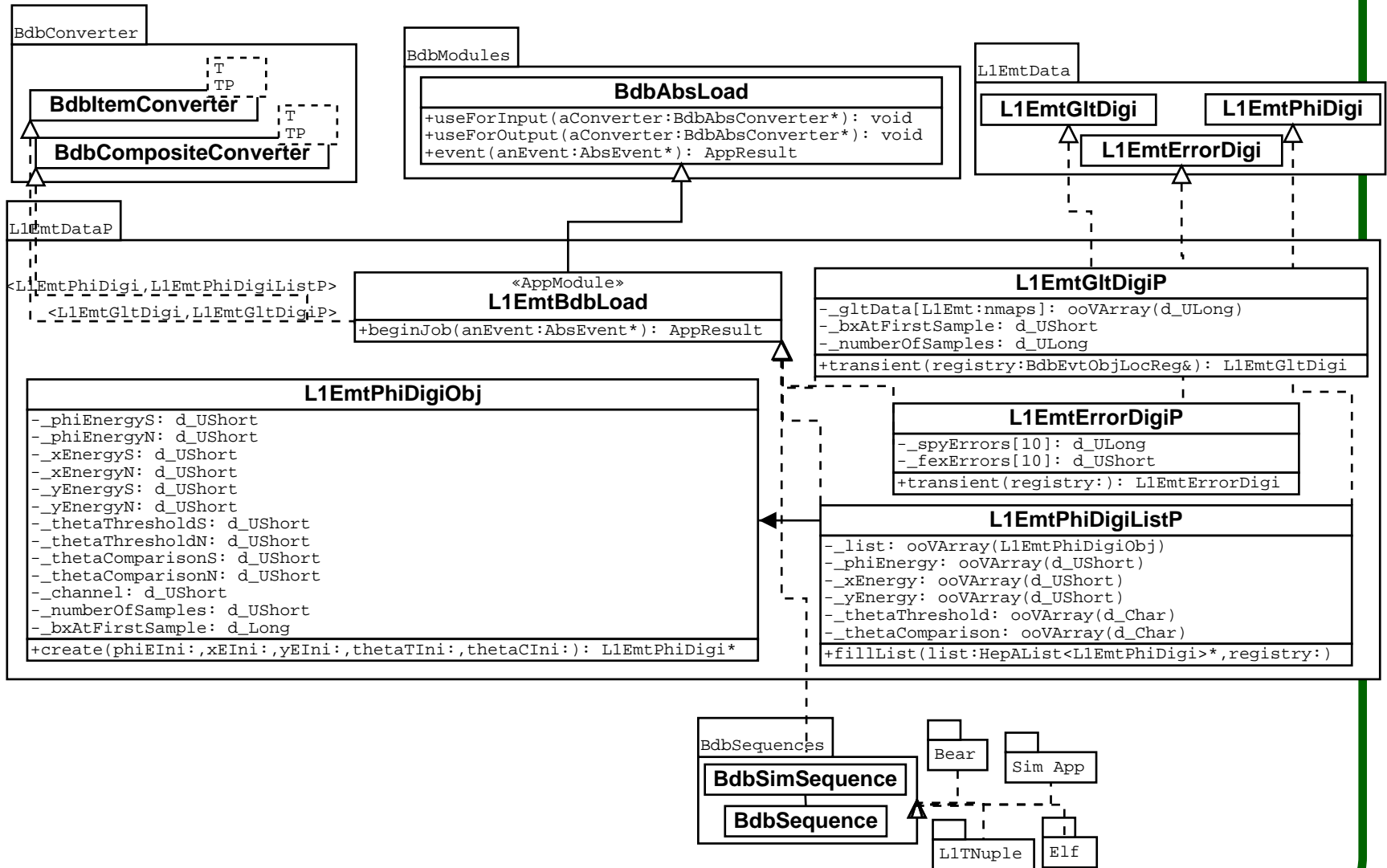
`unsigned short cableErrors[10]`

L1EmtDataCheck

Simple module that makes histos of the length and channel of the phi digi and prints it and the glt digi.

Future: Include error digi.

L1EmtDataP



L1EmtDataP

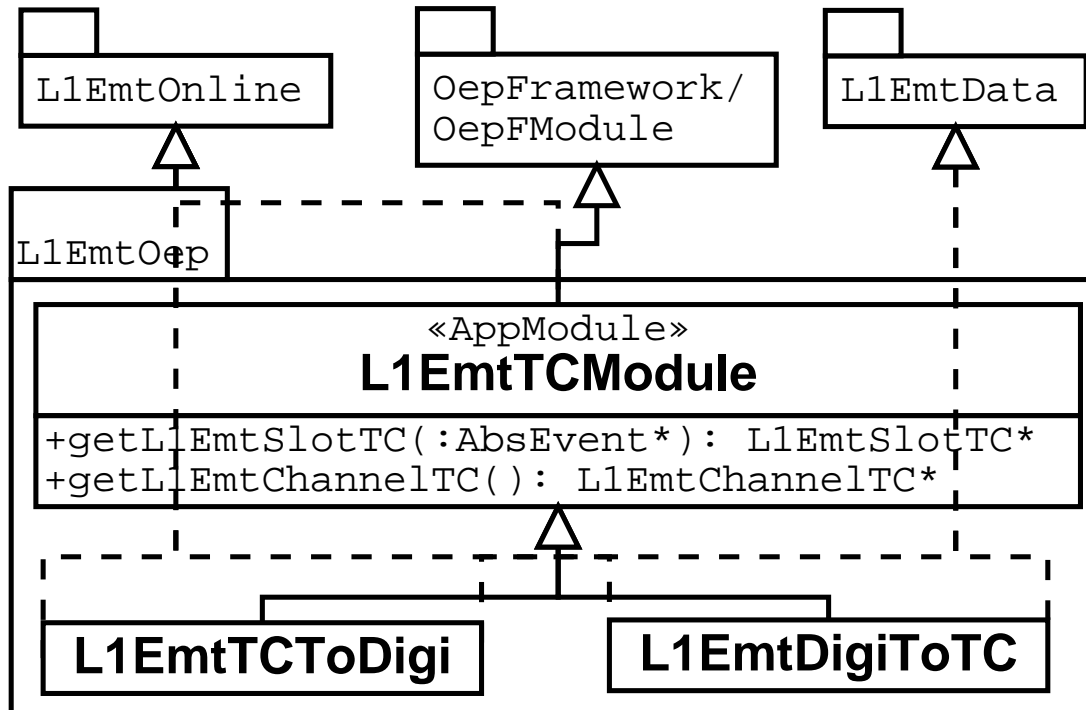
L1EmtBdbLoad

- In `beginJob()` creates `BdbItemConverters` for the Glt and Error DigiPs.
- Creates a `BdbCompositeConverter` for the `PhiDigiListP`.
- Passes these to the (inherited) `useForInput()` and `useForOutput()` depending on tcl params.
- `event()` function inherited from `BdbAbsLoad`.

DigiPs

- All take a digi in their constructor and have a member function to return it.
- `PhiDigiList` stores a list of `PhiDigiObjs` and one `ooVArray` for the energies/theta.
- Each `PhiDigiObj` stores start and extent of its energy.
- **Future:** Error Digi needs updating with CSER object (or just cable errors).

L1EmtOep : L1EmtTCModule



getL1EmtSlotTC:

- gets a scanner from the Event.
- does various checks.
- Makes a L1EmtSlotIterator for getting the channel TCs.
- returns the SlotTC.

getL1EmtChannelTC:

gets next channel TC from iterator and returns it.

L1EmtOep : L1EmtTCToDigi

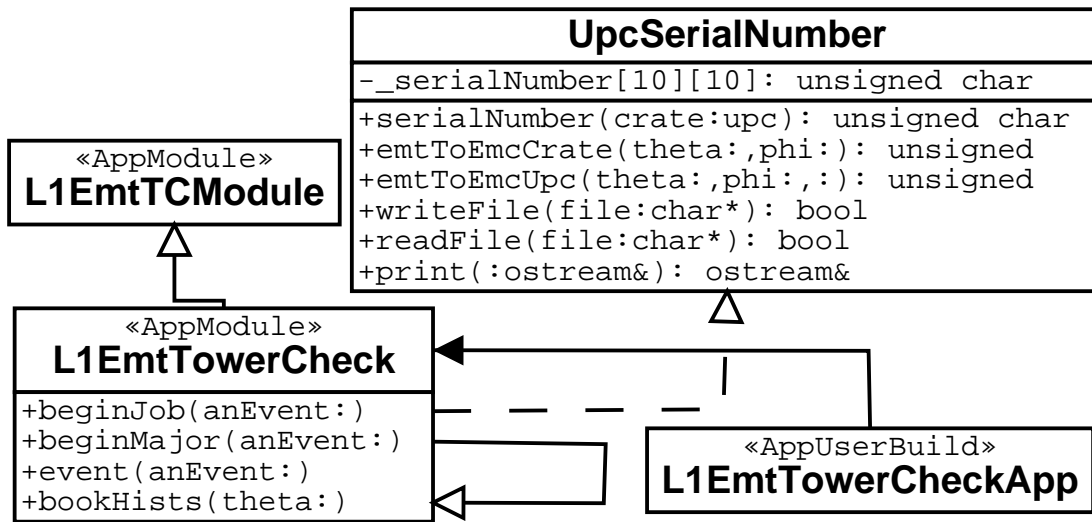
`event()` : Unpacks TC with `getL1EmtSlotTC` and makes 1 Glt and Error Digi per event then loops over channel TCs with `getL1EmtChannelTC` to make HepAList of 40 Phi Digis.

Digi	TC
<code>GltDigi::_gltData</code> <code>GltDigi::_bxAtFirstSample</code> <code>GltDigi::_numberOfSamples</code>	<code>SlotTC::gltData()</code> <code>SlotTC::clk60AtFirstGltSample()</code> Hardcoded to 32
<code>ErrorDigi::_spyErrors</code> <code>ErrorDigi::_fexErrors</code> <code>ErrorDigi::_cser</code>	<code>SlotTC::spyErrors()</code> <code>SlotTC::fexErrors()</code> <code>SlotTC::cser()</code>
<code>PhiDigi::_numberOfSamples</code> <code>PhiDigi::_bxAtFirstSample</code> <code>PhiDigi::_phiEnergy</code> <code>PhiDigi::_xEnergy</code> <code>PhiDigi::_yEnergy</code> <code>PhiDigi::_thataThreshold</code> <code>PhiDigi::_thetaComparison</code> <code>PhiDigi::_channel</code>	Hardcoded to 16 <code>SlotTC::clk60AtFirstChannelSample()</code> <code>ChannelTC::phiEnergy</code> <code>ChannelTC::xEnergy</code> <code>ChannelTC::yEnergy</code> <code>ChannelTC::theta()&0xff</code> <code>ChannelTC::theta() >> 8</code> <code>ChannelTC::channel()</code>

L1EmtOep:L1EmtDigiToTC

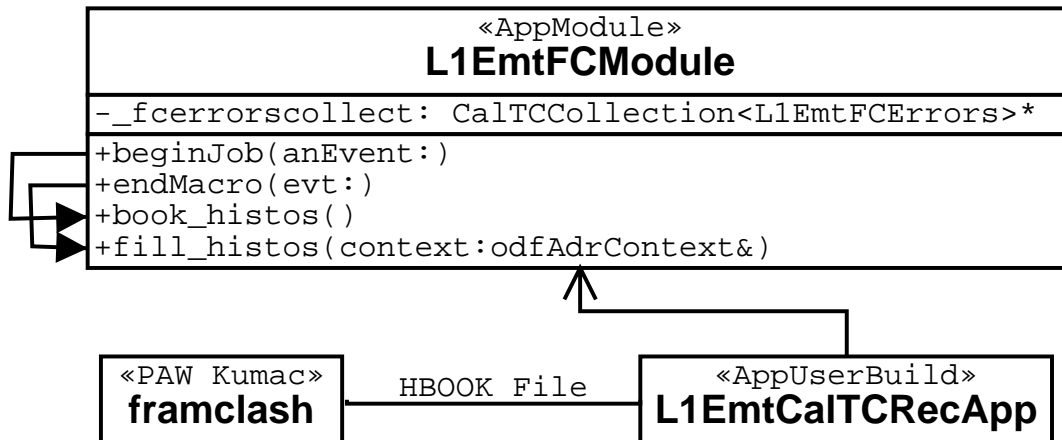
- Assigns some space for a new `L1EmtSlotTC` (not using `odfSimpleArena`)
- Puts the stuff from the Digi into the SlotTC
- Calls `gltSuppress()` from SlotTC to zero-suppress this data.
- Does error digi - but not CSER **Future**: add.
- Extracts PhiDigis from the event and iterates over them.
- Copies channel TC's info into PhiDigis.
- Expands slot TC to correct size
`sTC→setExtent()`;
- Makes a handle.
- Adds TC to event using `odfSimEventBuilder` with this handle.

L1EmtOep:L1EmtTowerCheck(App)



- Runs on TCs produced during serial number runs.
- Prints out “energies” - UPC Serial Numbers.
- Compares to those in [UpcSerialNumber](#) - reports mismatches.
- Histograms Raw Serial Numbers, mismatch Errors, Cable Errors and Theta bits.

L1EmtCalTCRecApp, L1EmtFCModule



- Runs on the event level in a Frameclash calibration receiving a stream of TCs.
- Retrieves [CalDataTCs](#) using the scanner.
- Finds [L1EmtFCError](#) Type ID.
- Constructs a collection [_fcerrorscollect](#).
- Histograms Cable Errors vs frame for each synch delay, tpb.
- and Safe frames (for each delay) (and split and bad cables.)
- Displayed by [frameclash.kumac](#)

L1EmtOep: The Rest - Useful - Maybe?

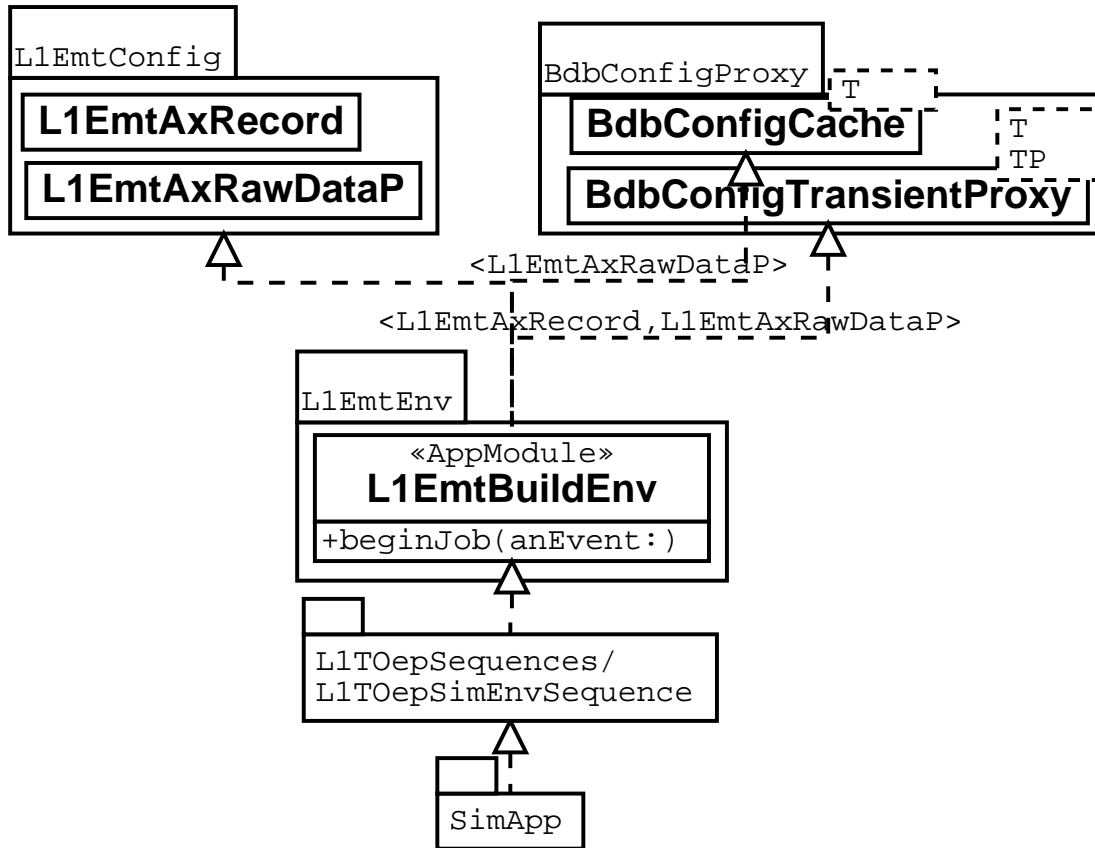
- [L1EmtTCCheckApp](#): Checks a number of properties and histograms.
- [L1EmtTCPrintApp](#): Runs L1EmtTCModule with verbose.
- [L1EmtGltCheckApp](#): Compare data from L1EmtSlotTC to L1GltTC.
- [L1EmtZeroCrossApp](#): “Used to study whether the timing offset from the fir zero-cross to the peak is correct.”
- [L1EmtEmcCorruptionApp](#): “Runs on data from the special EMC-EMT test runs in which the EMC writes out the tower sums it sends to the EMT.”
- [L1EmtRingMaskApp](#): “Studies the effect of masking out the inner rings of the endcap”
- [towercheck.kumac](#): for output histos from L1EmtTowerCheckApp.

L1EmtOep: The Rest - Outdated?

- [L1EmtTCPrintModule](#): Can use TCModule with verbose
- [L1EmtMonModule](#), [emtmon.kumac](#), [emtshmon.kumac](#):
Superseded by L1TFMon
- [L1EmtTCReceiveModule](#): Functionality available in TCModule
- [L1EmtRetriggerModule](#): No longer a problem?
- [L1EmtCompatScanSequence](#): Compares old (V1) TCs with new
- [L1EmtTCV1Module](#): Old TCV1
- [L1EmtTCALMonModule](#): Superseded by L1EmtTCCheck
- [L1EmtTestReceiverApp](#) and [L1EmtTestXTCMonitorApp](#): apps for outdated modules.
- [L1EmtDataSizeModule](#): histograms extent and data over various thresholds.
- [L1EmtTestFakeDataApp](#) ([L1EmtDataFake](#) + [L1EmtCompareDigisModule](#)): makes "fake" data to test TCtoDigi (and back) conversion.
- [tcalmon.kumac](#): ?

Future: Remove ?(!)

L1EmtEnv



- Puts EMT configuration info into “global dictionary” **gbIPEnv**
- Does this in exactly the same way as:
`L1EmtConfigInstantiate::proxies(lfdProxyDict* theDict)`
(See Phil’s talk)
- Only presently used to enable **L1EmtSim** to see Ax configuration info.

L1EmtEnv:Future

- Add Fcx and Fx Record - Easy
- Use single L1EmtEnv object?
- Reuse code shared with
L1EmtConfig/L1EmtConfigInstatiate?

Monitoring

See Also:

<http://www.slac.stanford.edu/BFROOT/www/Detector/Trigger/emt/software/monitoring/monitoring.html>

L1TFMon

- Fills all L1 histograms used for monitoring.
- Run on [Trickle Stream](#) for Fast Monitoring.
- [XTCApp](#) can be run off line and is routinely for [cron job](#).
- L1TFMon extracts Digis and passes them to L1TFMonEmt. Could pass event (?)
- **Future:** Add capability for selcting on L3 lines for MIP/Bhabha peak and more.

L1TMonTools

FastMonHTML

- Contains HTML displayed on [JAS](#) pages.
- Individual [XML](#) files for each displayed histo: sets display options.
- XML can be automatically generated by JAS.

FastMonAuto

- Can automatically compare live and reference histos.
- Outputs error messages via [CMLOG](#) and more.
- Presently only “EMT Output map M” compared.
- **Future:** Add 2D occupancy plot.

New tag can be installed yourself: See web page.

JAS

- Run in IR2 by “navigator” .
- Run from home (office) with
`/nfs/bbr-srv02/bfdist/Production/bin/SunOS5/RemoteBabarJas`
- Can test on hbook files or own trickle stream:
see web page.
- All references stored in single HBOOK file at
`/nfs/bbr-srv02/u4/Monitoring/References/merged/LiveFastMon.hbook`
Instructions for updating on web page
- All histograms made in Level3 and LiveFastMon
available (even if not plotted).

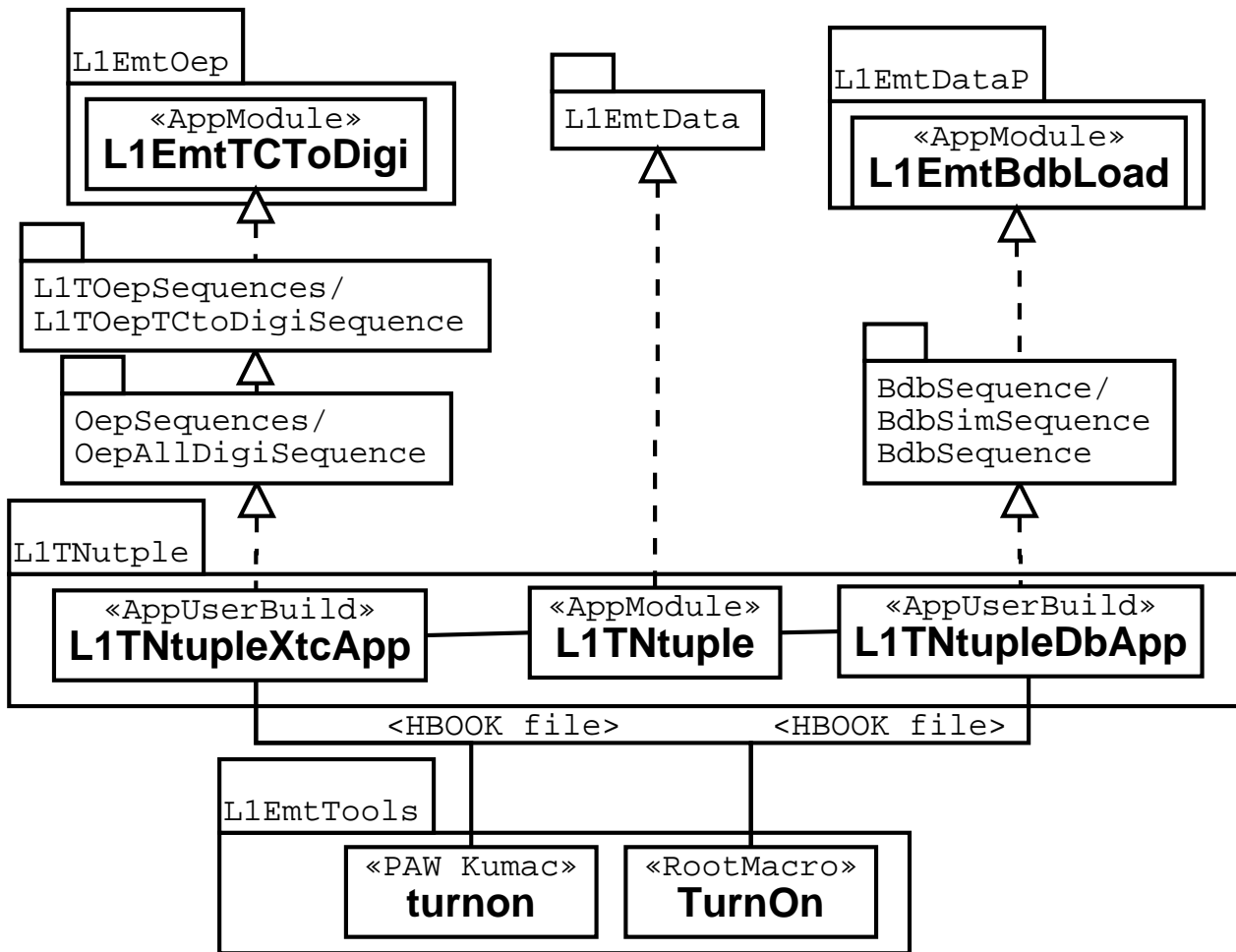
The Tower Mask Histogram

- Only EMT histogram from [IR2LiveL3](#).
- 2D Histo of present mask from DB.
- Superimposed on Occupancy plot in JAS.
- [L3Trigger](#) package calls 2 modules from [L1EmtConfig](#)
[L3InitSequence](#) adds [L1EmtConfigLoader](#) to load EMT proxies into configDict
[L3TAddL3Modules](#) adds [L1EmtConfigMonitor](#) which gets the [L1EmtAxRecord](#) from [configDict](#) and fills a histogram (see Phil's talk).

L1EmtTools:QuickMon(L3)

- Runs on Histograms dumped out at end of Level3 or FastMon job.
- Available immediately in emergency.
- Perl Script that runs kumac.
- Interactively chose date/recent runs etc.
- Produces small ps file for home viewing.

L1TNtuple



L1TNtuple

- Puts Digi information into ntuples.
- Includes useful quantities from other subsystems (eg EMC, GLT).
- L1TNtuple-variables.tex in package documents variables.
- Web Page at:
http://www.slac.stanford.edu/BFROOT/www/Detector/Trigger/operations/ntuple_instr_readme.html
- Most easily run with l1*ntp.prod scripts.

L1EmtTools

TurnOn.C

- Root Macro that runs on L1TNtuple output.
- Produces “Turn On” curves of trigger efficiency versus energy, Waveforms, Energy Spectra and more.
- **Future:** Fully document macro code.
- **Future:** Add further functionality.

Misc:

- [EmtCables.txt](#): List of UPC/TPB cables, numbers and status.
- [odfCopyReleaseEmt](#): Script which copies EMT libraries from Build area to dataflow area when new release built.

my Future

- Change CserRawData in ErrorDigi to cableErrors.
- Put this in DataP and DigiToTC
- Add L3Lines to Monitoring: for MIP/bhabha peaks and more.
- Clean up/comment all code mentioned here.
- Clean out L1EmtOep.
- Add functionality to TurnOn and document.
- Add Fx/Fcx to Env.
- Add 2D occupancy to Automatic Monitoring.