INNOWACYJNA GOSPODARKA
NARODOWA STRATEGIA SPÓJNOŚCI

UNIWERSYTET JAGIELLOŃSKI W KRAKOWIE

FNP
Fundacja na rzecz Nauki Polskiej

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ ROZWOJU REGIONALNEGO

INTERNATIONAL PHD PROJECTS IN APPLIED NUCLEAR PHYSICS AND INNOVATIVE TECHNOLOGIES

This project is supported by the Foundation for Polish Science – MPD program, co-financed by the European Union within the European Regional Development Fund

TRBv2 readout and unpacking system

Grzegorz Korcyl (grzegorz.korcyl@gmail.com)
International PhD Studies in Applied Nuclear Physics and Innovative Technologies UJ, Kraków

Plan

1. Full readout setup
2. Simplified readout setup
3. Triggering HPTDC
4. Data structure
5. TRBv3
6. HLD files
7. Unpacker
8. Unpacker structure
9. Running the unpacker

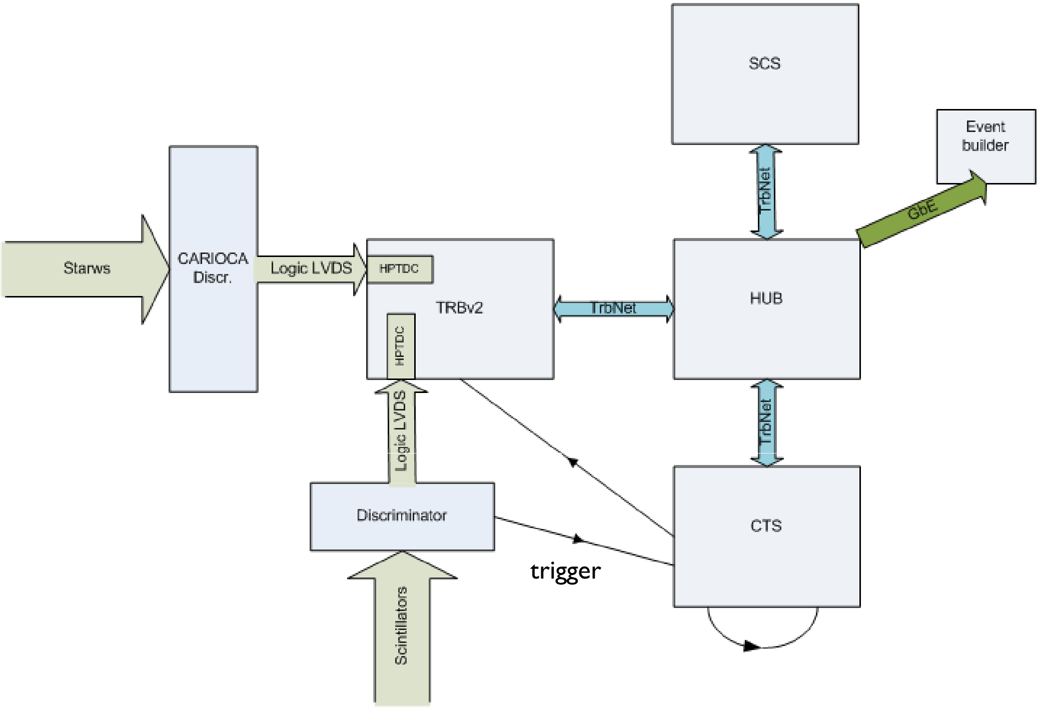
Full Readout Setup

- Expandable up to 16 data sources
- GbE (50MBps) uplink to Event Builders

- TRBv2
  - 4 HPTDC x 32 channels
  - 2Gbps optical link

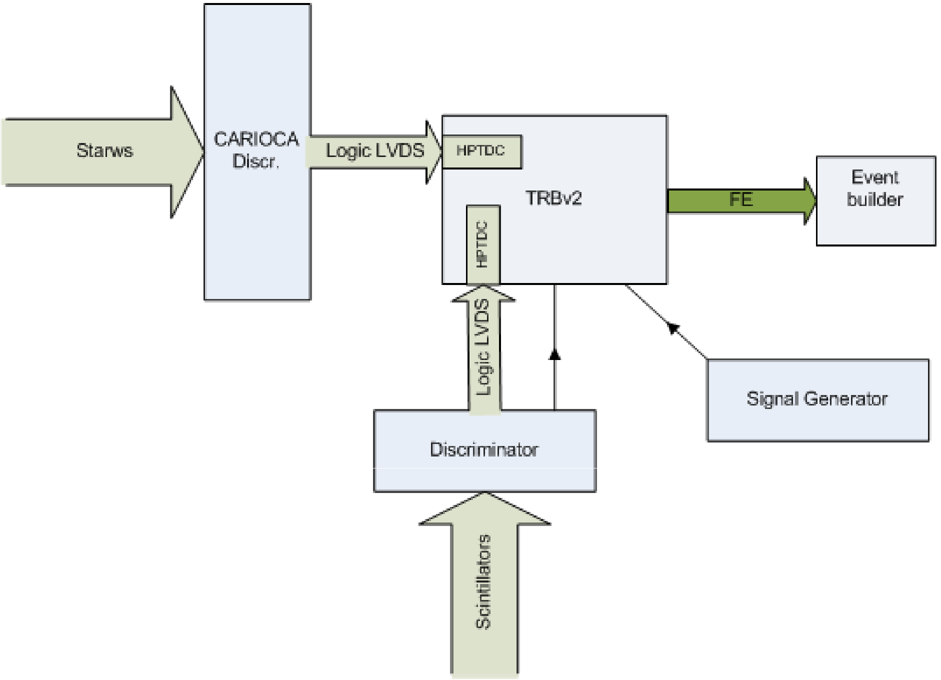
- Triggering controlled by CTS Addon
  - Triggered from external source
  - Multiple trigger inputs gating
  - Self trigger at specified rate

- Rates in idle state (1x TRB, 1xTDC):
  - Trigger: 135kHz
  - Uplink: 25MBps



Simplified Readout Setup

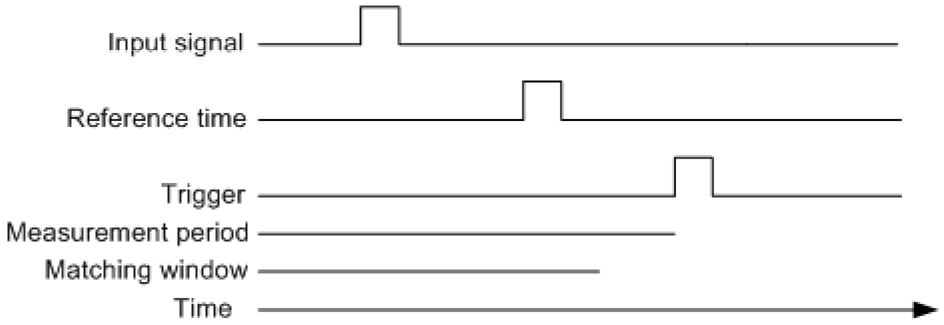
- 4 HPTDC x 32 channels
- FE (8MBps) uplink to Event Builder
- Needs external trigger source
- Rates
  - Trigger: 160kHz (theoretical)
  - Uplink: 8MBps



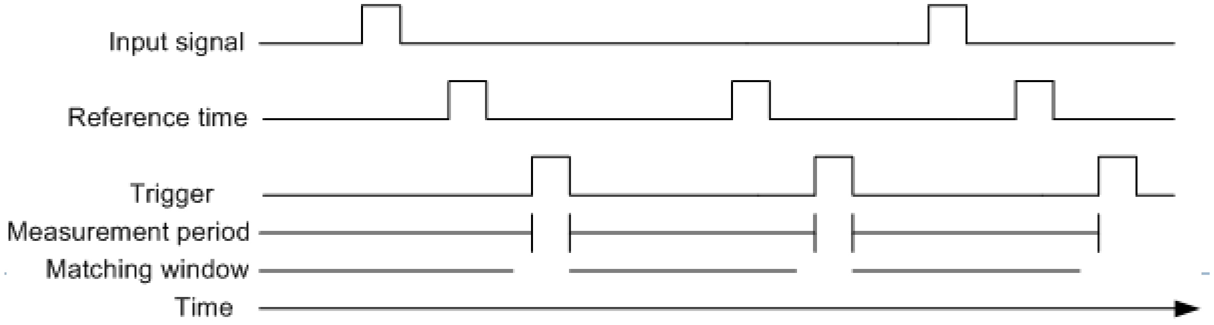
Triggering HPTDC

- External trigger input on TRBv2
- Signal connected to each TDC on 32nd channel and treated as reference time
- TDC triggered with a configurable delay (0xc0 [31;24])
- File configurable parameters

- Asynchronous trigger

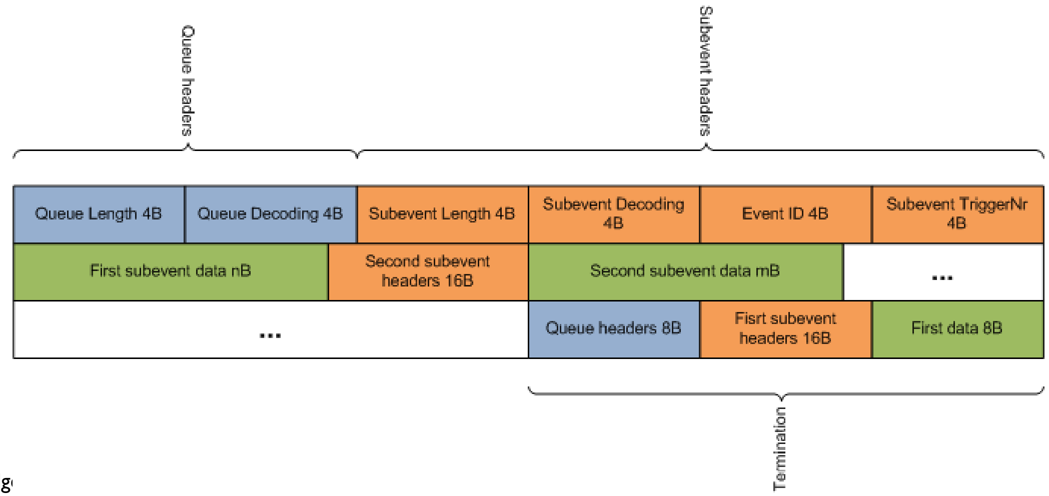


- „Triggerless"



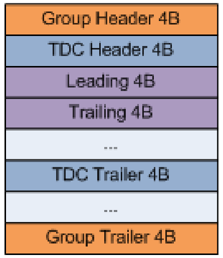
Data Structure

- „Full" setup
  - Collects and encapsulates data on several levels
  - Addressing needed
  - Deeper unpacking needed
  - Larger overhead:
    - Minimal empty event:
      - □ 40B Queue headers
      - □ 16B Subevent headers
      - □ 12B HUB headers
      - □ 8B + 4 (8B + n*8B) TDC headers + data (single edg
      - □ 8B + 4 (8B + n*4B) TDC headers + data (pairing mode)
      - □ = 108B
    - Addition from CTS ~20B

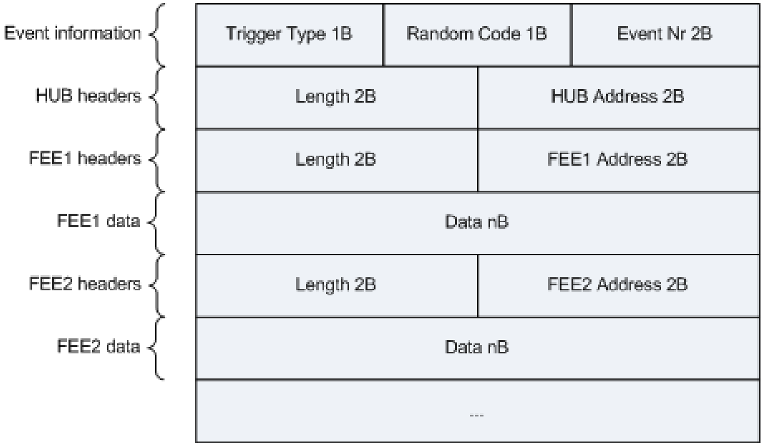


Queue Length 4B	Queue Decoding 4B	Subevent Length 4B	Subevent Decoding 4B	Event ID 4B	Subevent TriggerNr 4B
First subevent data nB		Second subevent headers 16B	Second subevent data mB		...
...			Queue headers 8B	Fisrt subevent headers 16B	First data 8B

- „Simplified" setup
  - Data comes from only one source
  - Smaller overhead:
    - Minimal empty event:
      - □ 8B Endpoint headers
      - □ 8B + 4 (8B + n*8B) TDC headers + data (single edge)
      - □ 8B + 4 (8B + n*4B) TDC headers + data (pairing mode)
      - □ = 48B



Group Header 4B
TDC Header 4B
Leading 4B
Trailing 4B
...
TDC Trailer 4B
...
Group Trailer 4B

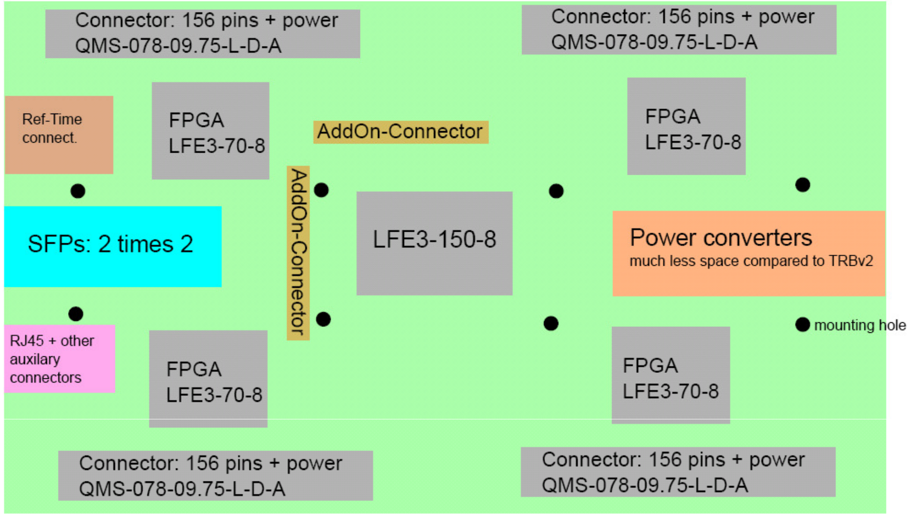


Event information	Trigger Type 1B	Random Code 1B	Event Nr 2B
HUB headers	Length 2B	HUB Address 2B	
FEE1 headers	Length 2B	FEE1 Address 2B	
FEE1 data	Data nB		
FEE2 headers	Length 2B	FEE2 Address 2B	
FEE2 data	Data nB		
	...		

TRBv3

- Key features:
  - 4x Lattice ECP3 70
  - 1x Lattice ECP3 150
  - 4x SFP
    - 2x GbE duplex links
    - 2x TrbNet links
  - 4x 156 pin connectors with power supply
    - Input lines for TDC-in-FPGA
    - Connectors to small Addon Boards
  - 1x regular Addon connector

TRBv3 layout proposal

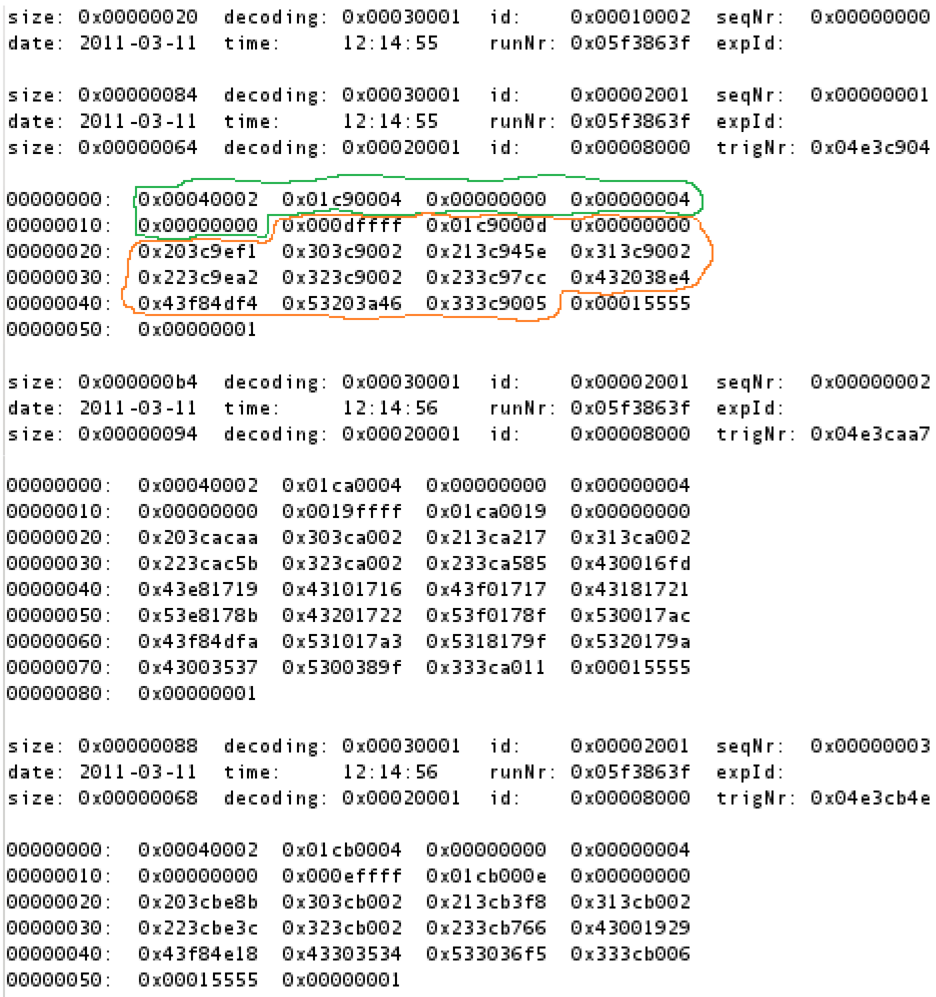


- Configurable number of TDC channels
- Extendable number of data transmitters

- TDC in FPGA
  - ~ 6ps time resolution
  - 17ps in Hi-Res mode by HPTDC

HLD File

- Binary file with saved events
- Subevents structure
- Tools:
  - Daq_anal – decodes data to ASCII
  - HYDRA – heavyweight framework for full unpacking
  - Unpackers – lightweight ROOT library unpacker



```
size: 0x00000020  decoding: 0x00030001  id:     0x00010002  seqNr:  0x00000000
date: 2011-03-11  time:     12:14:55    runNr:  0x05f3863f  expId:

size: 0x00000084  decoding: 0x00030001  id:     0x00002001  seqNr:  0x00000001
date: 2011-03-11  time:     12:14:55    runNr:  0x05f3863f  expId:
size: 0x00000064  decoding: 0x00020001  id:     0x00008000  trigNr: 0x04e3c904

00000000:  0x00040002  0x01c90004  0x00000000  0x00000004
00000010:  0x00000000  0x000dffff  0x01c9000d  0x00000000
00000020:  0x203c9ef1  0x303c9002  0x213c945e  0x313c9002
00000030:  0x223c9ea2  0x323c9002  0x233c97cc  0x432038e4
00000040:  0x43f84df4  0x53203a46  0x333c9005  0x00015555
00000050:  0x00000001

size: 0x000000b4  decoding: 0x00030001  id:     0x00002001  seqNr:  0x00000002
date: 2011-03-11  time:     12:14:56    runNr:  0x05f3863f  expId:
size: 0x00000094  decoding: 0x00020001  id:     0x00008000  trigNr: 0x04e3caa7

00000000:  0x00040002  0x01ca0004  0x00000000  0x00000004
00000010:  0x00000000  0x0019ffff  0x01ca0019  0x00000000
00000020:  0x203cacaa  0x303ca002  0x213ca217  0x313ca002
00000030:  0x223cac5b  0x323ca002  0x233ca585  0x430016fd
00000040:  0x43e81719  0x43101716  0x43f01717  0x43181721
00000050:  0x53e8178b  0x43201722  0x53f0178f  0x530017ac
00000060:  0x43f84dfa  0x531017a3  0x5318179f  0x5320179a
00000070:  0x43003537  0x5300389f  0x333ca011  0x00015555
00000080:  0x00000001

size: 0x00000088  decoding: 0x00030001  id:     0x00002001  seqNr:  0x00000003
date: 2011-03-11  time:     12:14:56    runNr:  0x05f3863f  expId:
size: 0x00000068  decoding: 0x00020001  id:     0x00008000  trigNr: 0x04e3cb4e

00000000:  0x00040002  0x01cb0004  0x00000000  0x00000004
00000010:  0x00000000  0x000effff  0x01cb000e  0x00000000
00000020:  0x203cbe8b  0x303cb002  0x213cb3f8  0x313cb002
00000030:  0x223cbe3c  0x323cb002  0x233cb766  0x43001929
00000040:  0x43f84e18  0x43303534  0x533036f5  0x333cb006
00000050:  0x00015555  0x00000001
```

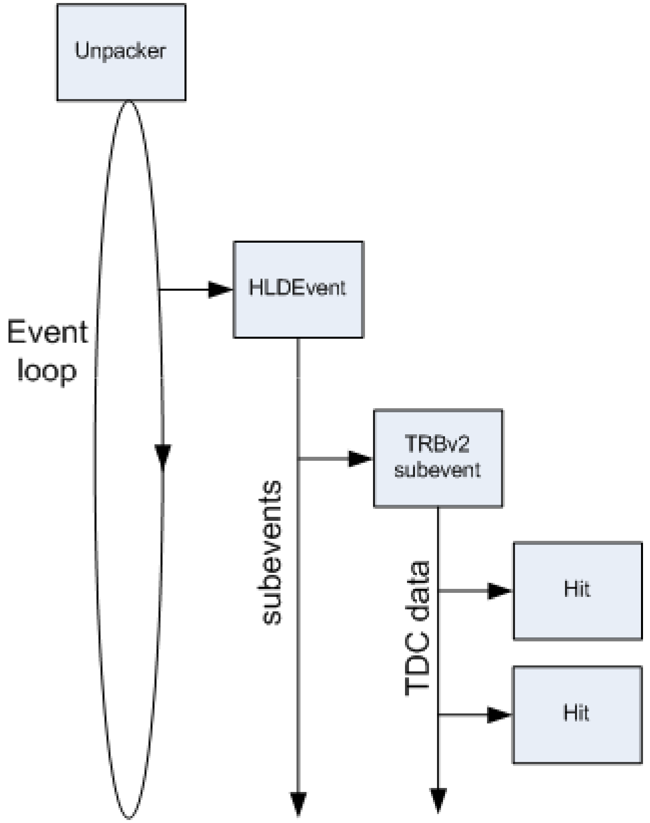
Unpacker

- Independent C++ software
  - Few source files and MakeFile needed to compile
- Designed as ROOT library to be loaded at startup
- Creates ROOT structures from HLD files

- Current limitations (but not hard to solve!):
  - Decodes only TDC data from TRBv2 subevents
  - Can decode data from one endpoint at a time

Unpacker Structure

- Unpakcer
  - Prepares ROOT tree
  - Iterates through all events in the file
  - Creates and executes HLDEvents
  - Fills the tree with decoded events
- HLDEvent
  - Contains
    - Information about event
    - List of Hits
  - Filters out event from specified endpoint
  - Decodes TDS data
  - Creates Hit objects
- Hit
  - Contains:
    - TDC number
    - Channel
    - Leading time
    - Trailing time / Width (mode dependent)



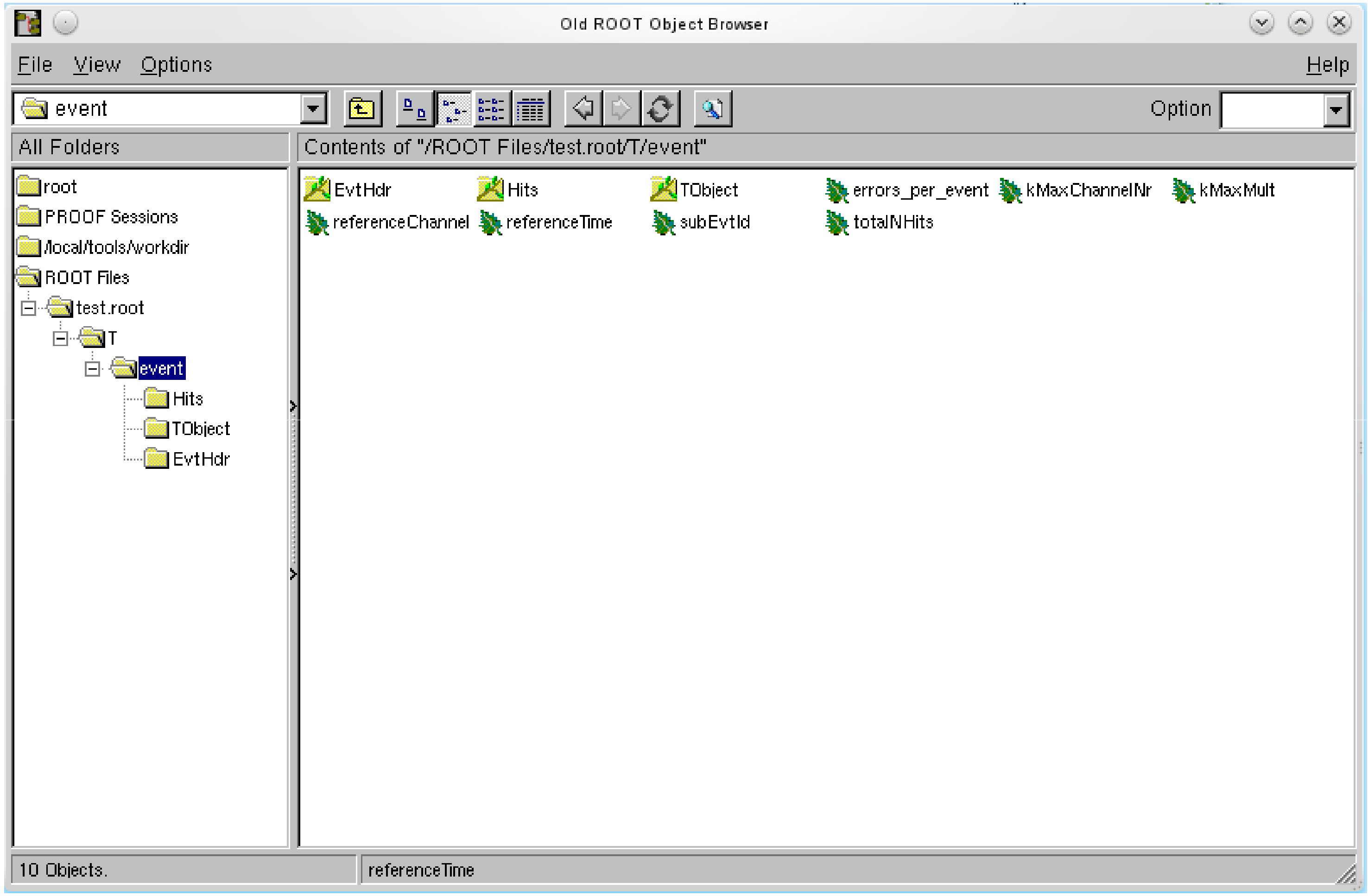
Running the Unpacker

- Compilation:
  - Simply make
  - No other dependencies
  - Creates libunpacker.so

- Load library to ROOT:
  - Start ROOT from a directory with rootlogon.c file
  - gSystem->Load(„path/to/unpacker/libunpacker.so”);

- Call the Unpacker:
  - Create an instance of Unpacker
  - Unpacker(const char* hld, Int_t nEvt, Int_t subEvtId, Int_t refChannel)

Running the unpacker

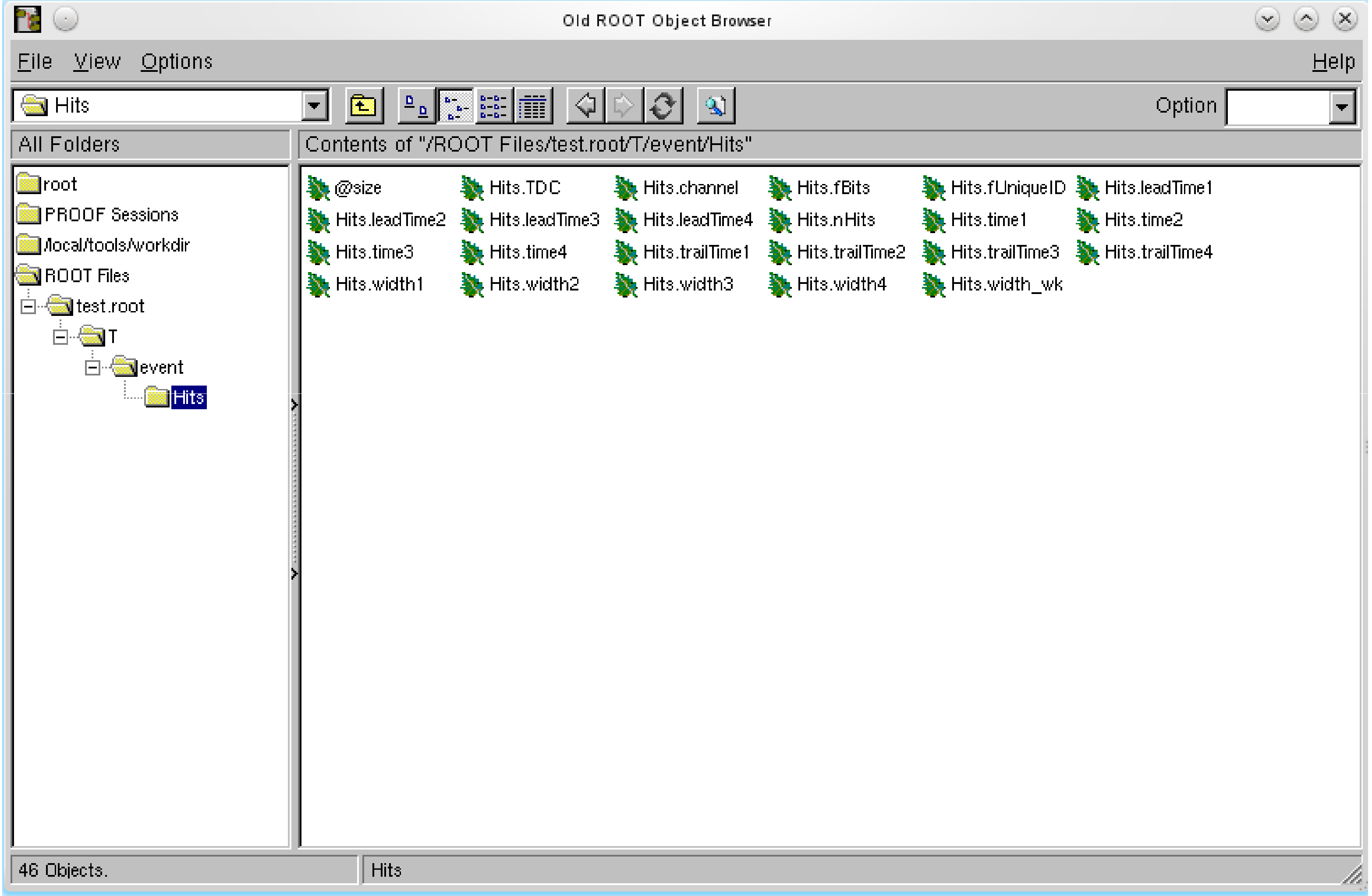


Contents of "/ROOT Files/test.root/T/event"

10 Objects.

referenceTime

Running the Unpacker



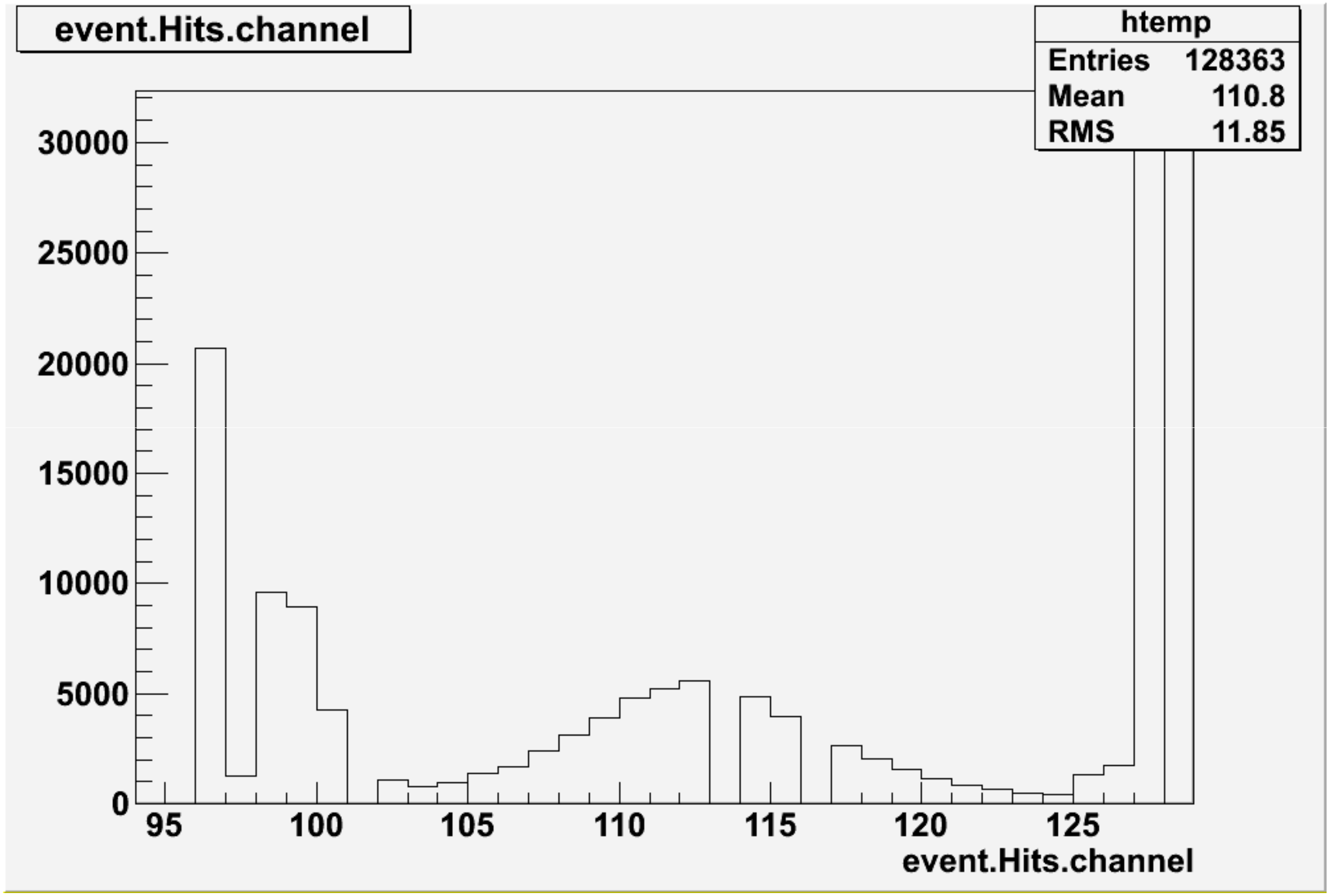
Contents of "/ROOT Files/test.root/T/event/Hits"

- root
- PROOF Sessions
- /local/tools/workdir
- ROOT Files
  - test.root
    - T
      - event
        - Hits

@size	Hits.TDC	Hits.channel	Hits.fBits	Hits.fUniqueID	Hits.leadTime1
Hits.leadTime2	Hits.leadTime3	Hits.leadTime4	Hits.nHits	Hits.time1	Hits.time2
Hits.time3	Hits.time4	Hits.trailTime1	Hits.trailTime2	Hits.trailTime3	Hits.trailTime4
Hits.width1	Hits.width2	Hits.width3	Hits.width4	Hits.width_wk	

46 Objects. Hits

Running the unpacker


event.Hits.channel
htemp
Entries 128363
Mean 110.8
RMS 11.85
event.Hits.channel