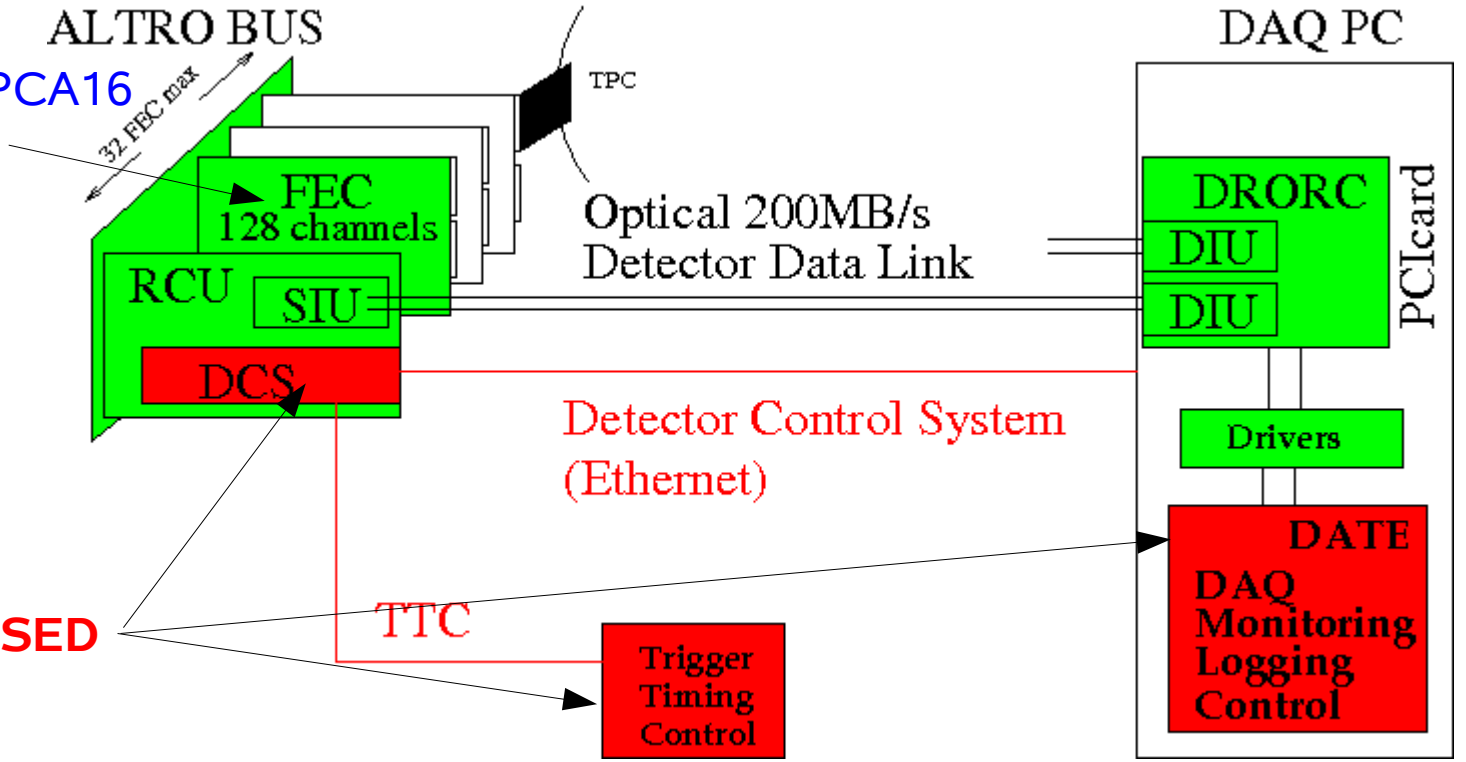
Readout of TPC with modified ALICE electronics

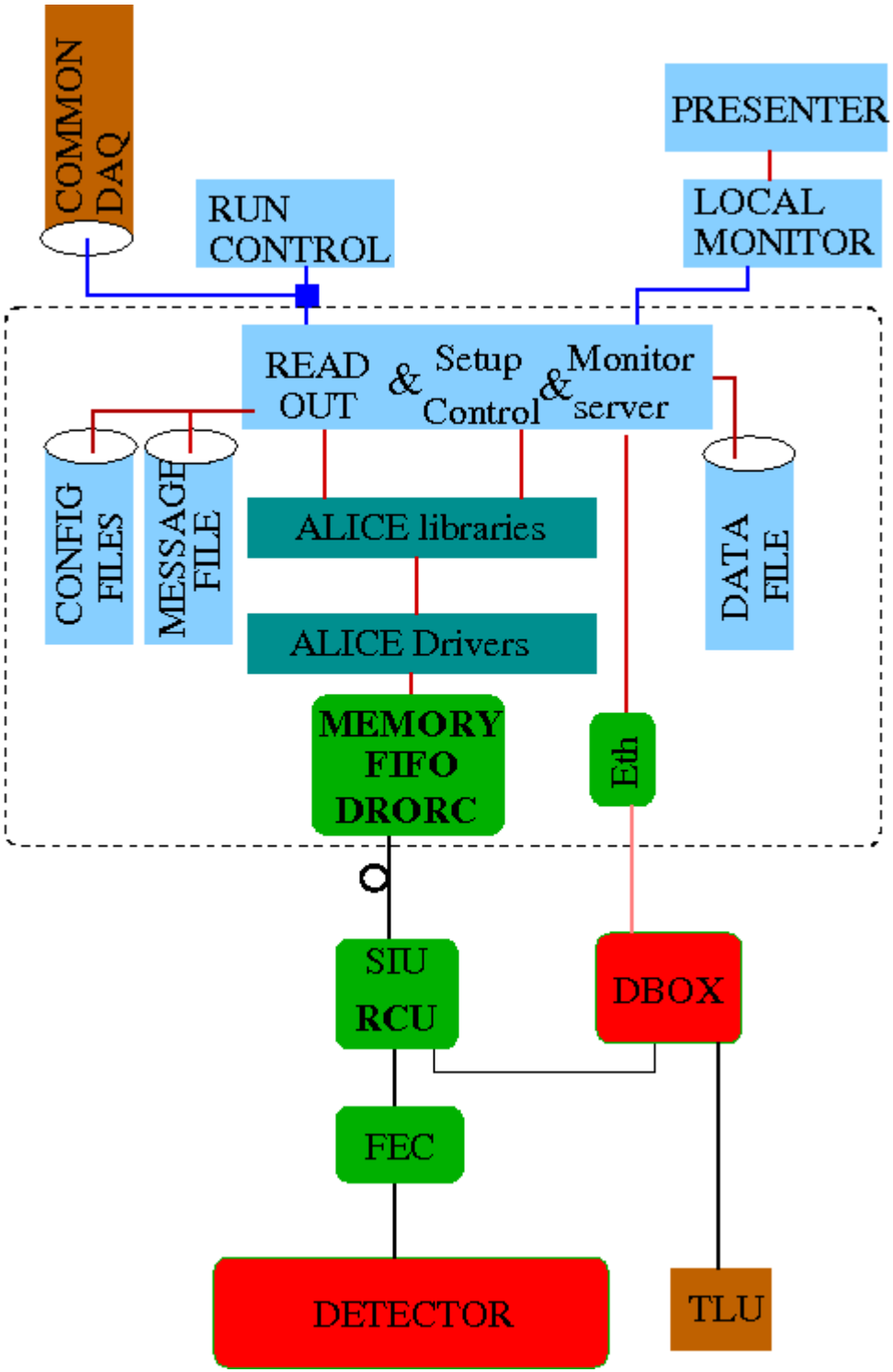
details of current version and pending items

ALICE overview

Modified:
PASA -> PCA16



New software based on homemade partly existing scheleton and ALICE drivers --->



Rudimentary Local Readout Software

ALICE drivers, libraries
Readout/setup server
Monitor server
Local run control (TCP/IP)
Local monitor (TCP/IP)
Histogram presenter
Configuration files: ASCII files
Message files: ASCII files
Data file: binary

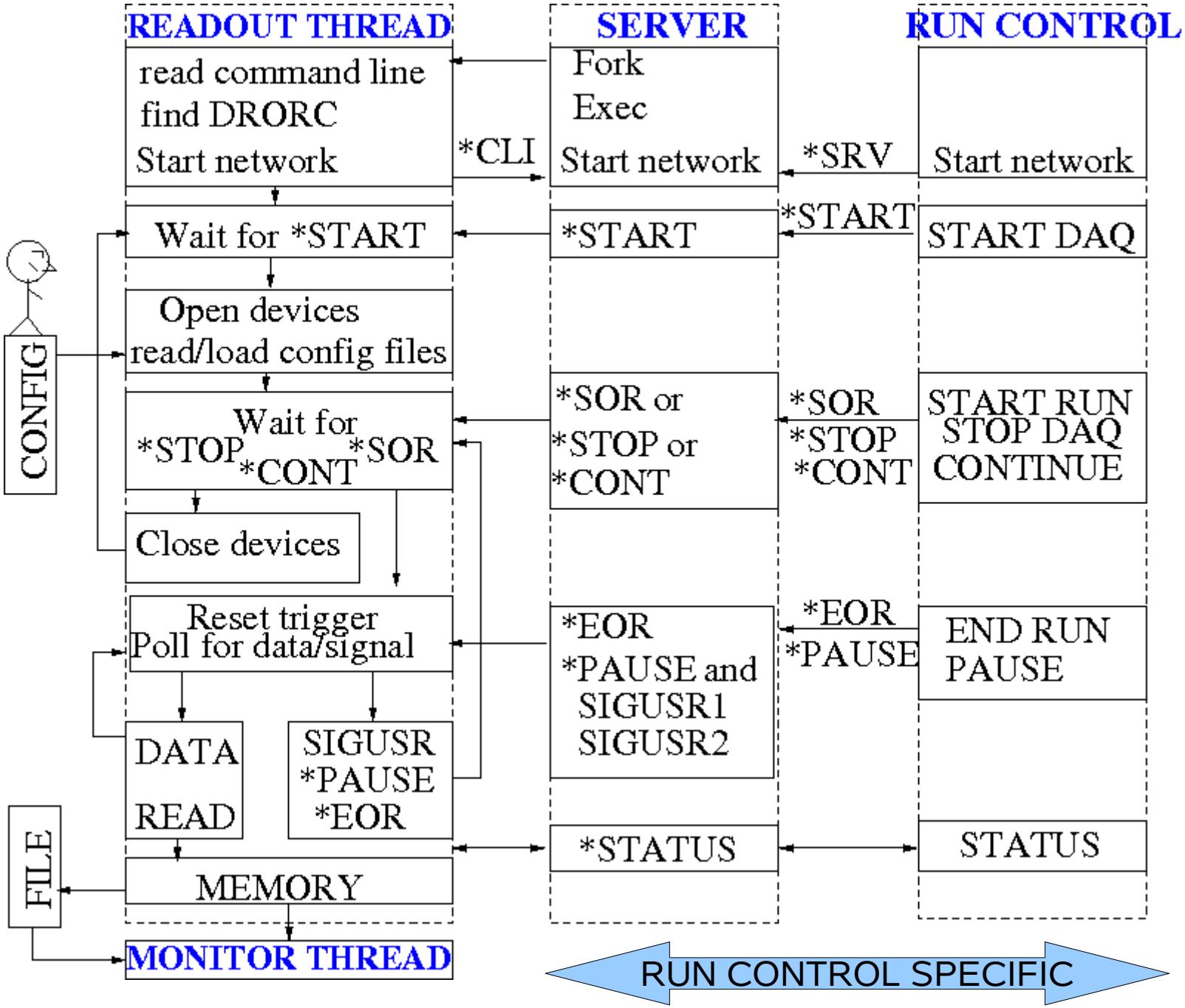
Trigger interface

DBOX: trigger/clocks to RCU, data
TLU: common trigger box

ALICE TPC Hardware

FEC: 8 16 channel ALTRO chips
RCU: Readout Control Unit <=32 FECs
DRORC: 2 RCUs/DRORC

???SLOW CONTROL???



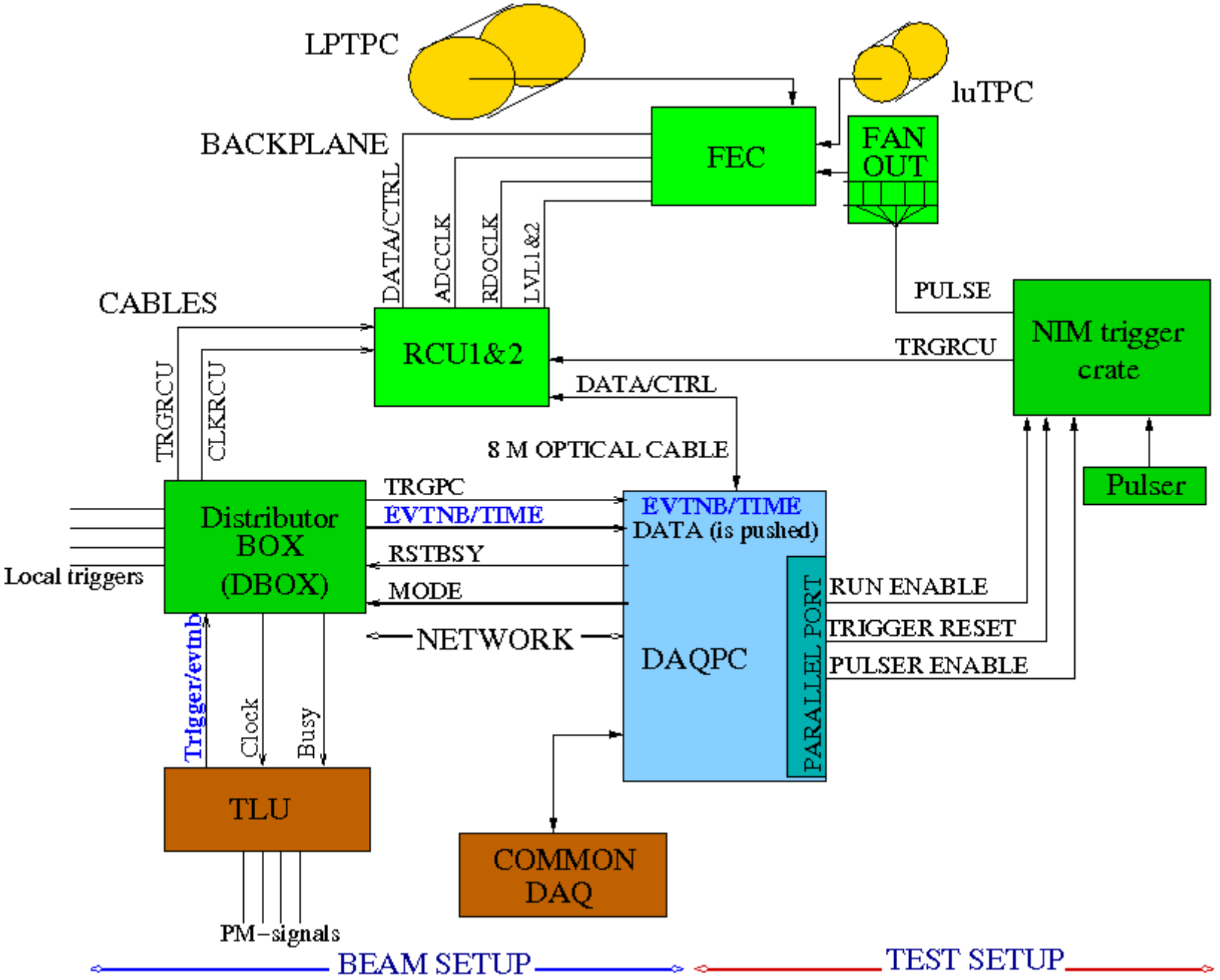
RUN HANDLING

Commands:
text strings

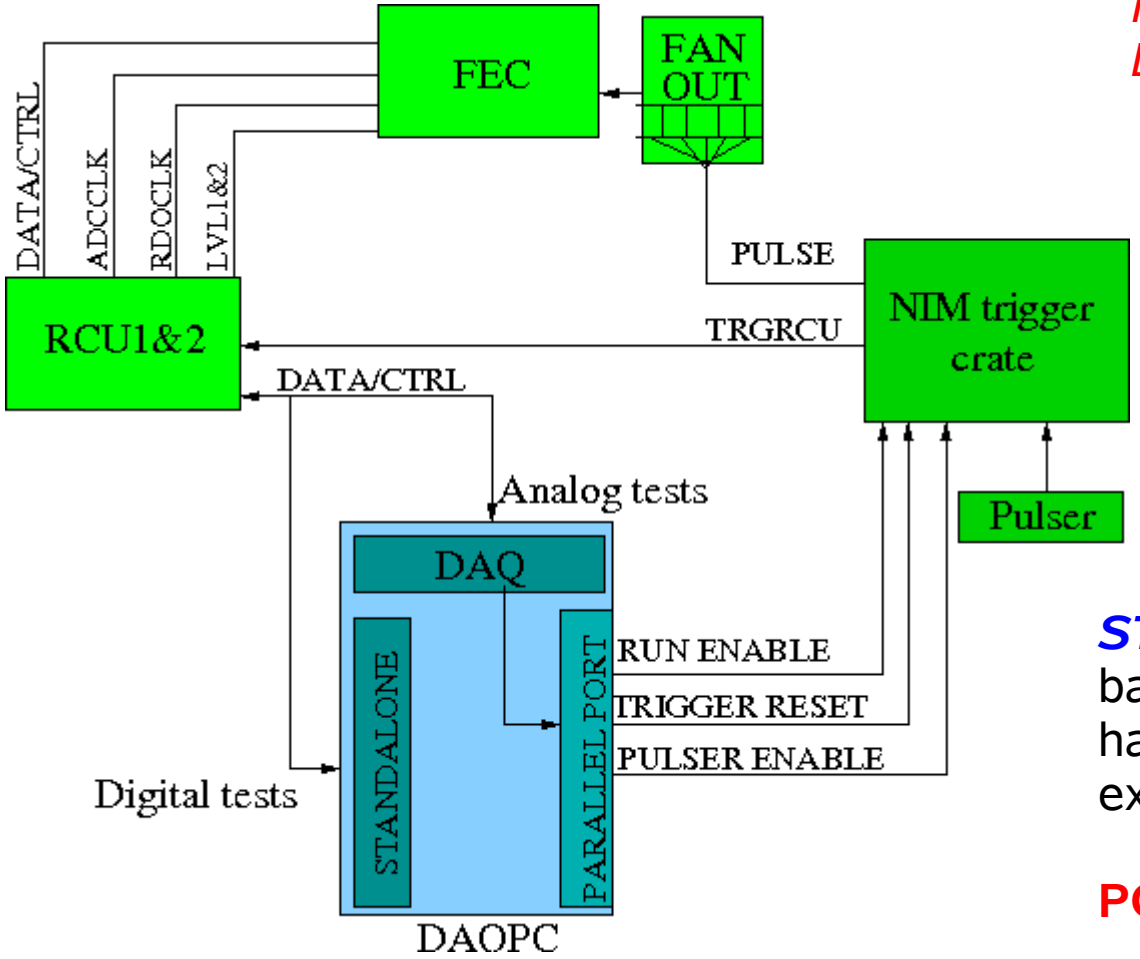
Protocol:
TCP/IP
Unix signals

Limitations:
one RCU,
2000 channels,
2MB/event
DAQ event buffer:
one event

TRIGGER, TEST, AND EVENT SYNCHRONIZATION



FUNCTIONALITY TEST OF FRONT END CARD



ANALOG TESTS

FANOUT card

distributes pulse: all, single,odd,even

manual switches

Investigate cross talk, noise, connections

DAQ used for readout

DIGITAL TEST

standalone tasks: write/read/compare individual registers

STATUS

basics exist/being produced

hardware/software will be finalized with experience from the test of prototype FEC

PCA16 rotated --> some delay!

Configuration files are ASCII files in the format NAME.TYPE <value>

files.cfg	definitions of locations and names of other files used
run.cfg	run information (run type,comment,pedestal files)
daq.cfg	general DAQ, updated each run (run number)
drorc.cfg	mapping RCU <-> DRORC and status of them
rcu-<rcuid>.cfg	for each RCU: list of active FEC/ALTRO
global-altro.<mode>	settings common to all ALTROs in all FECs
global-rcu.<mode>	settings common to all RCUs
	<mode>=physics,(pedestal(,test to be implemented))

Pending:
programming of PCA16
implement database and web interface?

Message files are ASCII files into which status/debug/information is logged

message log	information and debug output
run log	status of run,started,stopped, comment, ...
monitor log	status of monitor server

Pending:
must be cleaned up, is very messy at the moment
run parameter history

Example: settings of ALTRO

```
DPCFG.ZS_CFG        <value>
#                   bits 0-1:glitch filter
#                   bits 2-4:presamples excluded from zero suppression
#                   bits 5-6:postsamples excluded from zero suppression
#                   bit 7:enable zero suppression
ZSTHR.ZS_THR        <value>
#                   Zero suppression threshold
BFNPT.PTRG          <0-15>
#                   Pretrigger samples stored before trigger arrives
TRCFG.ACQ_END       <0-1000>
#                   Cycles to sample from trigger to acquisition end
#                   128 ch * 1000 cycles * 40 bits ALTRO /32 bits
#                   = 160 kbytes/FEC without zero suppression
DPCFG.BC1_CFG       <value>
#                   mode of operation = f(data in) – fixed pedestal
#                   where f is a function that inverts half of the input
#                   signals, due to PCA16 swap of outputs compared with ALICE PASA
```

Data files are binary files with the name readout-<runnb>_<filenb>.dat
A new file with a new <filenb> is created if MAX_FILE_SIZE is exceeded

Run records:

SOR	Start of run record.
EVENT	Event record.
RCU	Each event record consists of RCU data, each RCU has a header.
DATA	Event data of ALTRO format for each RCU.
EOR	End Of Run record.
BOF	If local data logging. If the file size exceeds a maximum size then the file is closed and a new file is opened and the first record in the new file is a Beginning Of File record.
EOF	If local logging. If the file size exceeds a maximum size then the file is closed and a new file is opened. The last record in the closed file is End Of File record.
POR	Pause run record
COR	Continue run record

DATA FORMAT – RUN RECORD EXAMPLES

START OF RUN FORMAT

Total length (exclusive)		
Header length (exclusive)		
Block identifier = BLOCK_SOR (=0x11111111)		
Data format version		
Run number		
Year	Month	Day
Hour	Minute	Second

END OF RUN FORMAT

Total length (exclusive)		
Header length (exclusive)		
Block identifier = BLOCK_EOR (=0x33333333)		
Number of events		
Year	Month	Day
Hour	Minute	Second

RAW EVENT FORMAT

RAW EVENT FORMAT (32-bit words)

Total event length (exclusive, added by software)
Header length (exclusive, added by software)
Block identifier = BLOCK_EVENT (=0x22222222) (added by software)
Software event number (incremented by software for each read event)
Hardware trigger number (read from distributor box)
Time stamp (read from distributor box)
RCU block length (exclusive, added by software)
RCU identifier (added by software)
RCU HEADER – 8 words
ALTRO DATA – N40 = # of 40 bit words = (N40*5)/4 32 bit words = N32
...
...
RCU block length (exclusive, added by software)
RCU identifier (added by software)
RCU HEADER – 8 words
ALTRO HW DATA – N40 40 bit words = (N40*5)/4 32 bit words = N32

ALTRO DATA are 40 bit words and in general not divisible by 4

N32 modulus 5 = 0

ALTRO WORD1 [31..0]	
ALTRO WORD2 [23..0]	ALTRO WORD1 [39..32]
ALTRO WORD3 [15..0]	ALTRO WORD2 [39..24]
ALTRO WORD4 [7..0]	ALTRO WORD3 [39..16]
ALTRO WORD4 [39..8]	

N32 modulus 5 = 2

ALTRO WORD1 [31..0]	
AAAAAAA	ALTRO WORD1 [39..32]

N32 modulus 5 = 3

ALTRO WORD1 [31..0]	
ALTRO WORD2 [23..0]	ALTRO WORD1 [39..32]
AAAA	ALTRO WORD2 [39..24]

N32 modulus 5 = 4

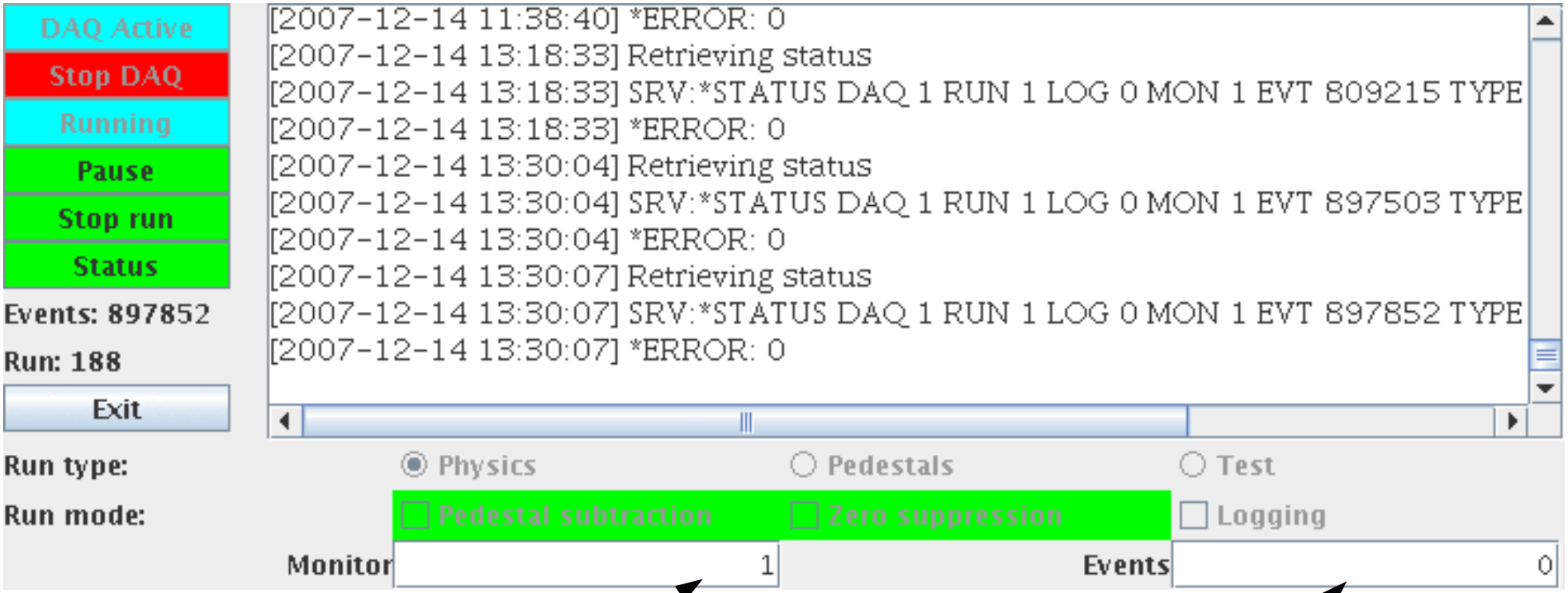
ALTRO WORD1 [31..0]	
ALTRO WORD2 [23..0]	ALTRO WORD1 [39..32]
ALTRO WORD3 [15..0]	ALTRO WORD2 [39..24]
AA	ALTRO WORD3 [39..16]

RAW DATA FORMAT

back linked data
need to find last data and position of last ALTRO word

N32 = 32 bit words from RCU

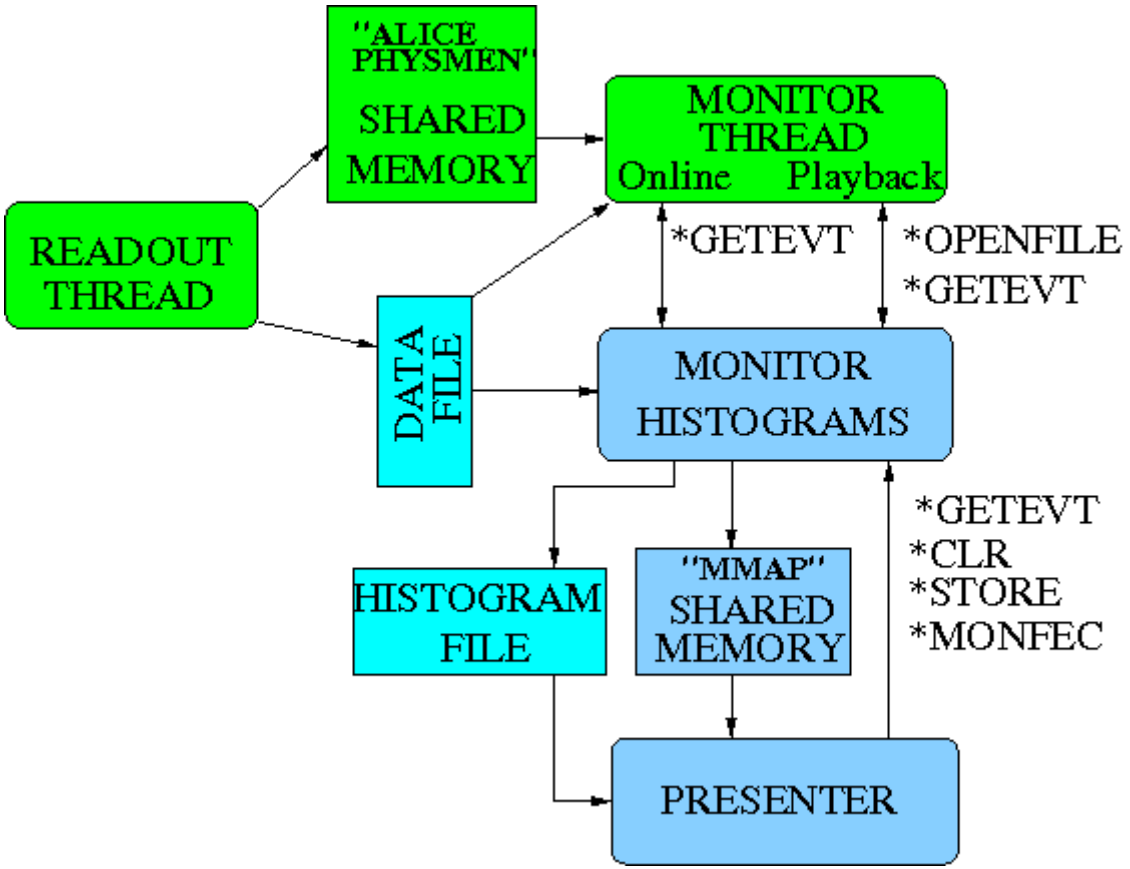
LOCAL RUN CONTROL (written in JAVA)



"Downscale factor" of events to monitor
0 = no events

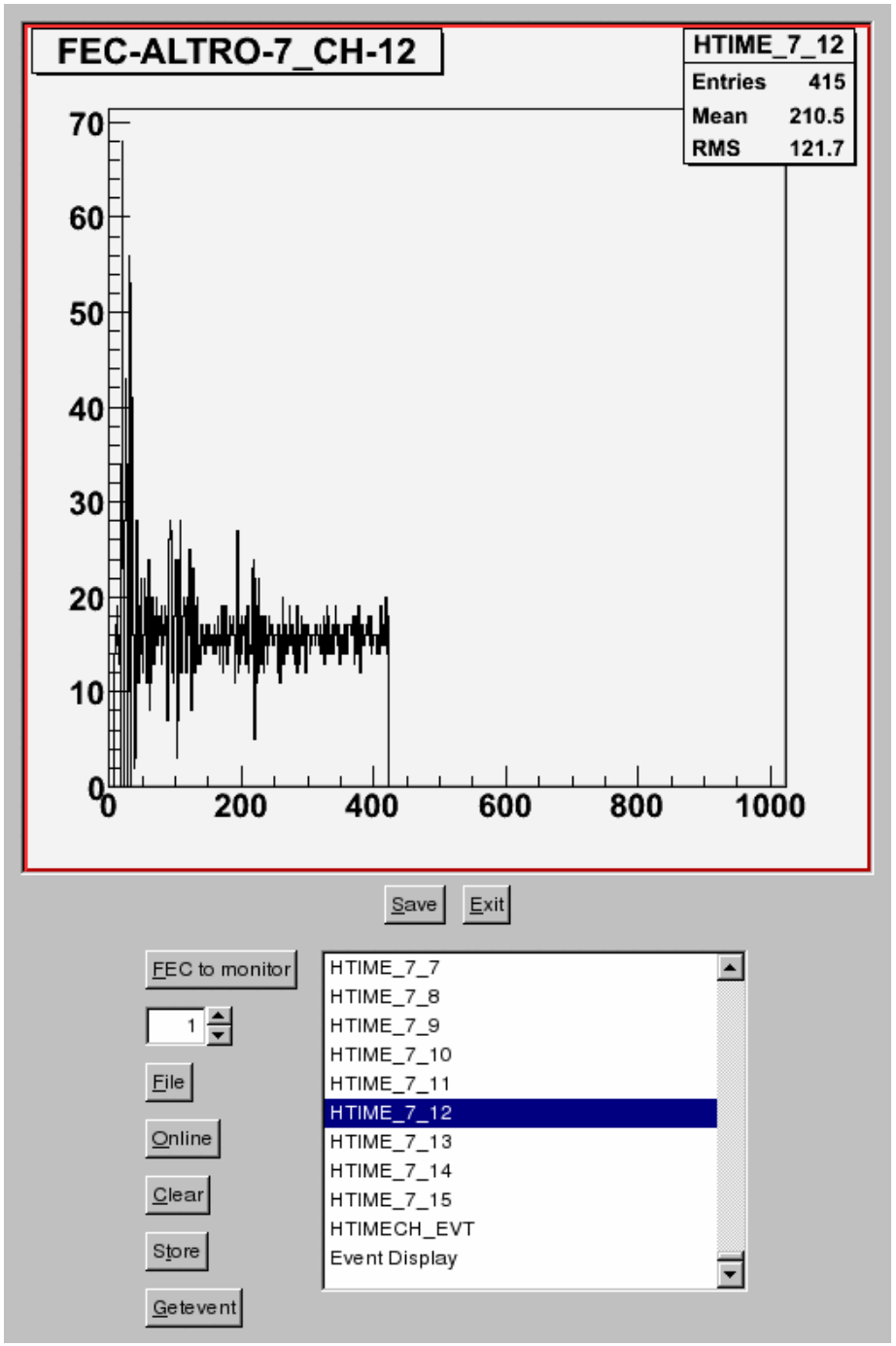
Events to take, 0 = infinity

MONITOR HANDLING (Written in C/C++ and ROOT)



Commands:
text strings

Protocol:
TCP/IP



HISTOGRAM PRESENTER (Written in ROOT)

FEC to monitor:
only one FEC (128 channels) at a time

Current histograms:
pulse heights accumulated/channel
time distribution for single events/channel
"event display" = channel vs cluster time

File: use histogram file
Online: use shared memory histograms
Clear: clear histograms
Store: store to histogram file
Getevent: Fill single event histograms

Save: current display as postscript file
Exit: exit program

STATUS

Tested:
parallel port as trigger interface
pulser as trigger
one unmodified ALICE FEC
one RCU
one DRORC

PENDING STEPS

Slow control **if any???**
RCU firmware update: *event number/clock frequency*
Trigger Interface (DBOX), event synchronization
Interface to common DAQ
Implement pedestal calculations/files
Implement test mode: test of modified front end cards
"Messy": needs to be cleaned up and improved, configuration database
TPC: connect to small TPC in Lund
Limitations: one RCU, 2000 channels, 2MB/event
DAQ event buffer: one event
High rate: hangs when saturated with triggers
PCA16: rotated

Schedule: depends on FEC tests....