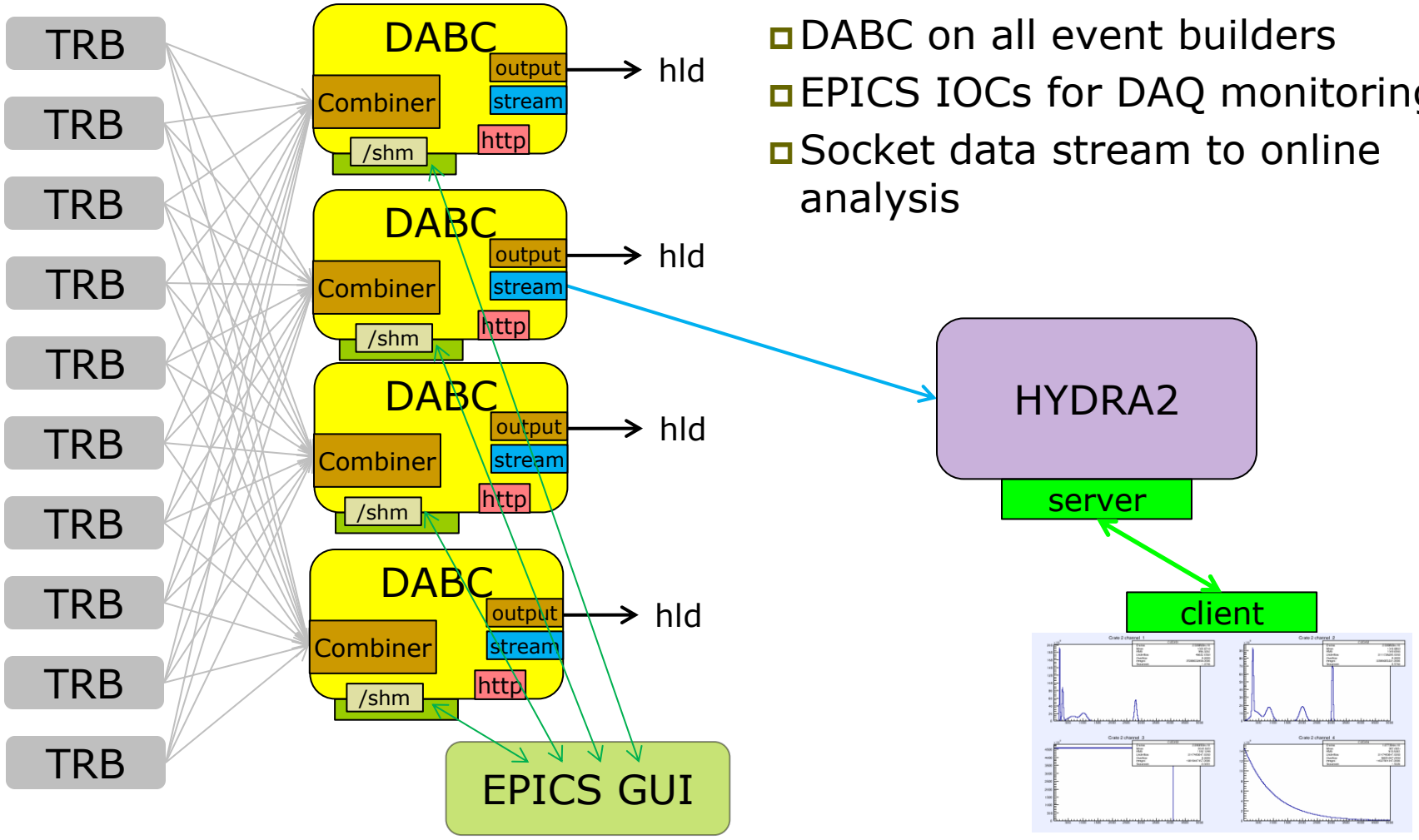
FPGA TDC calibrations in DABC

Sergey Linev

27.11.2014

HADES DAQ now

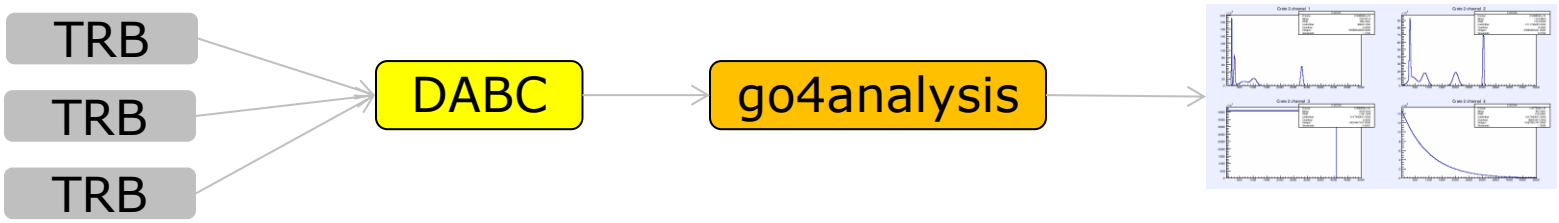


- DABC on all event builders
- EPICS IOCs for DAQ monitoring
- Socket data stream to online analysis

TDC calibrations in HADES now

- Steps to produce calibrations:
  - Setup calibration trigger, store HLD files
  - Produce calibration data from HLD files
  - Submit calibration into ORACLE
- Significant amount of data
  - ~2K per channel
- Takes a lot of time to produce

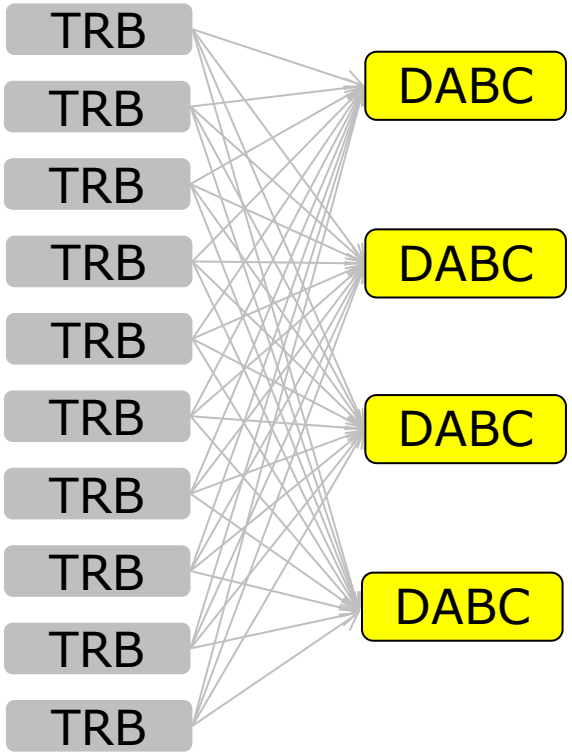
TDC calibrations in DABC/Go4 now



- Can be performed offline/online
  - typically 10^5 hits/channel required
- If enough statistic, 10 s to produce
- Calibrations can be stored in binary files
  - these files are used now in HYDRA

- Is it possible in HADES?

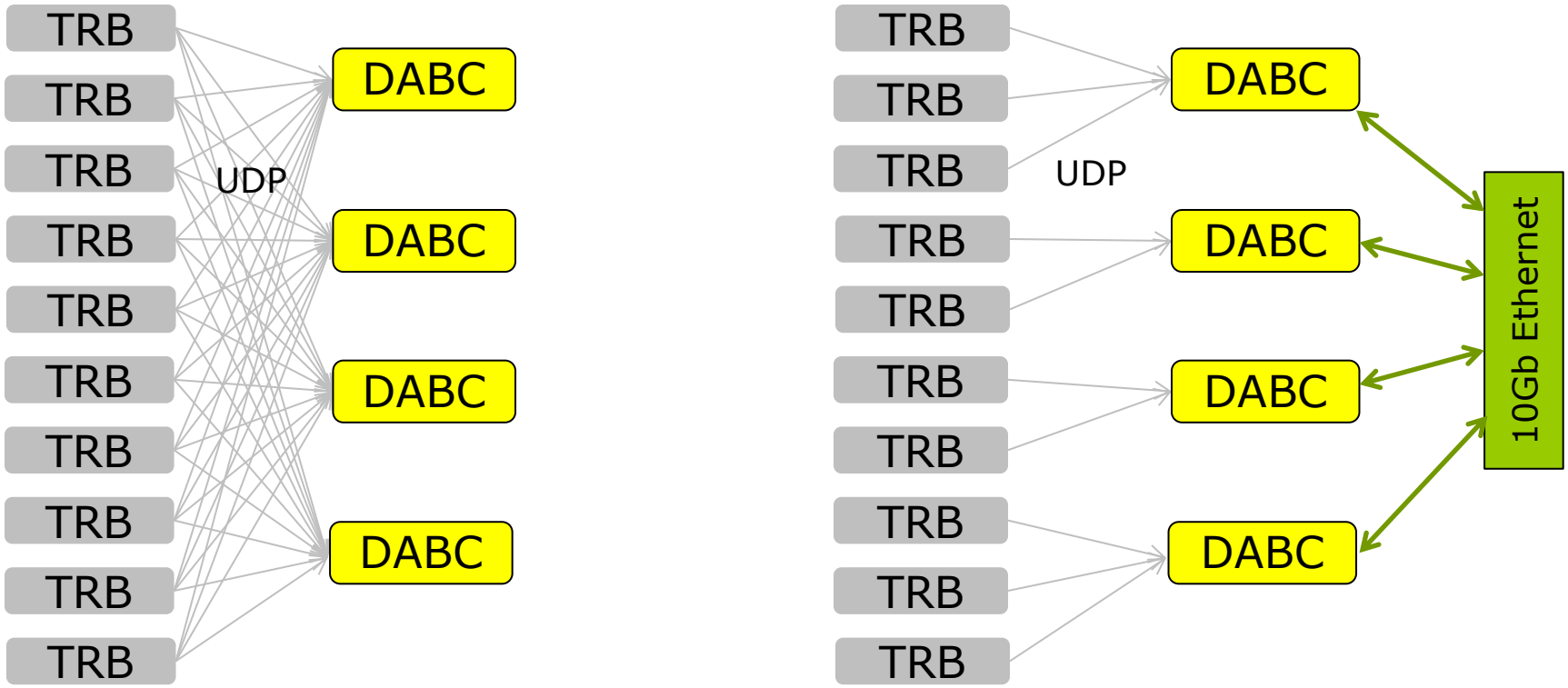
Problems with current setup



- All events equally distributed over DAQ nodes
- But one needs calibration events at one place
- One could forward calibration events to one event builder, but it can be overloaded
- Even if possible, still problem with calibration storage in ORACLE – it is too big
- All together will takes minutes before system ready

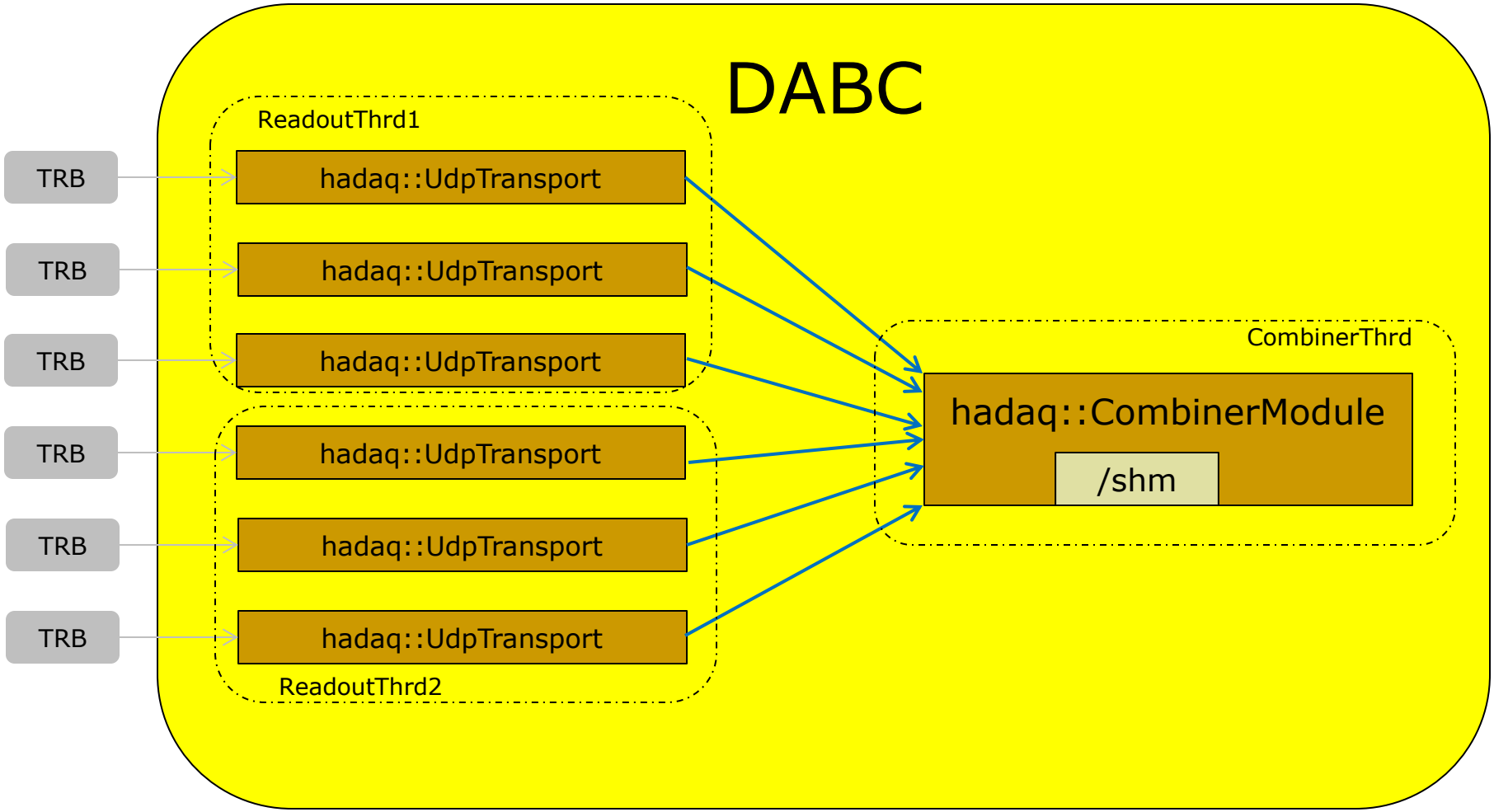
- Is there alternatives?

Change networking topology

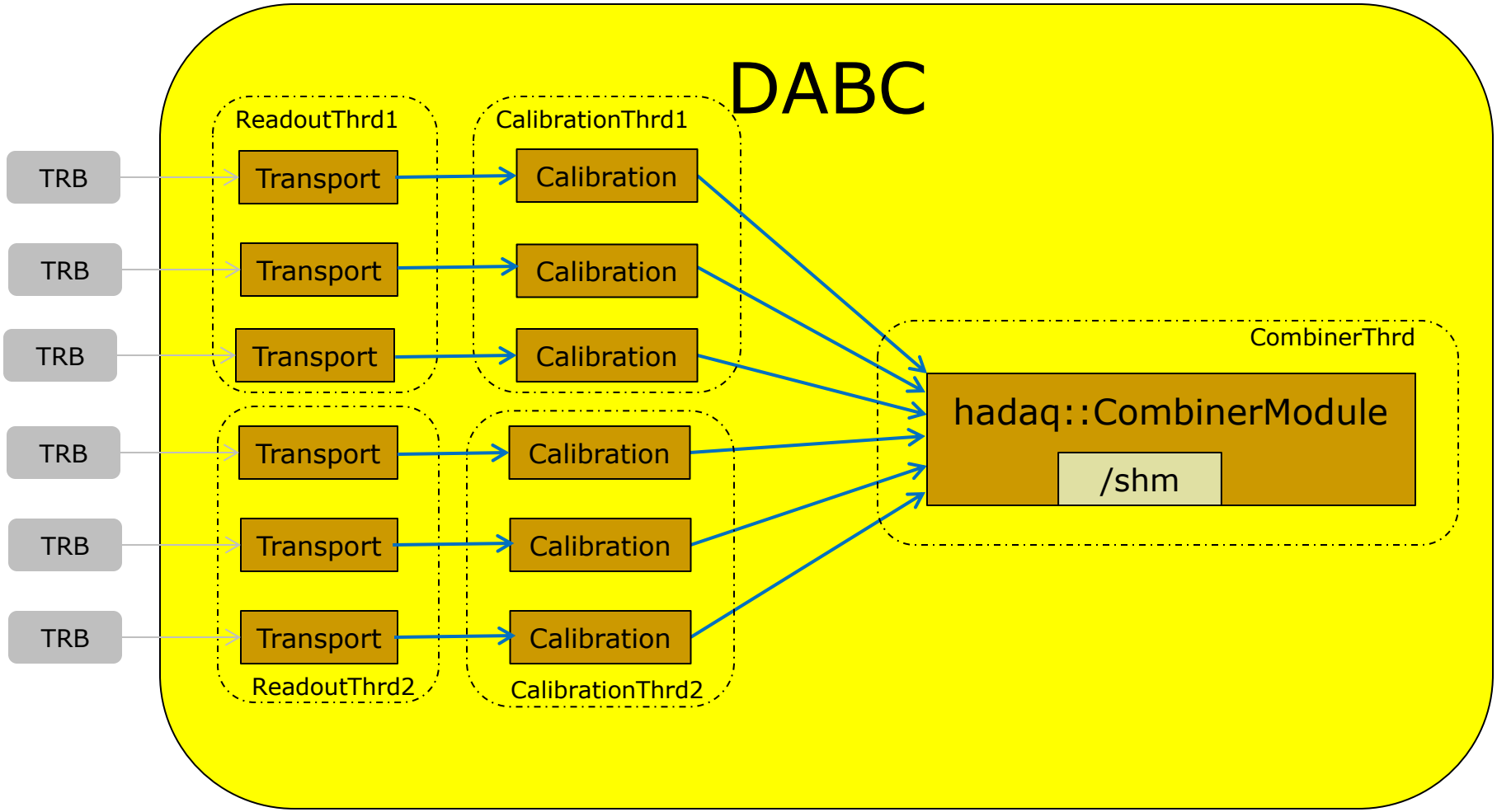


- All data from single TRB always send to same DAQ node
- Such DAQ node can generate and apply calibration immediately
- Event building performed afterwards with normal TCP over 10 Gb Eth

DABC node – how it works



DABC node with calibration ON



Calibration modes

- Automatic
  - each channel will be calibrated again after N hits, useful for test environment
- Semi-automatic
  - Statistic will be accumulated all the time, calibration will be generated by operator command from EPICS or just by time interval
- Static
  - Calibration loaded from the file, no any changes during DAQ run

Where to place calibrated values?

- Output of calibration is float value between 0 and 5 ns with about 1 ps precision – requires ~12 bits
- Hit message uses all 32 bits
- One could add extra message with calibrated values for next two hits (to save space)
- Approximate increase of raw data by 25%

```
epoch
   hit1 ch0
epoch
   hit2 ch 2
   hit3 ch 7
epoch
   hit4 ch 5
```

```
epoch
   fine 1+2
   hit1 ch0
epoch
   hit2 ch 2
   fine 3+4
   hit3 ch 7
epoch
   hit4 ch 5
```

Control and quality monitor

- Control and basic statistic
  - via EPICS
  - via web interface
- Histograms similar to current go4analysis
  - via web interface
    - see web-docs.gsi.de/~linev/js/
  - via Go4 GUI
    - native ROOT graphics

Calibration in DABC – more possibilities

- Volume of data for calibration triggers can be much higher than volume of normal events (one could generate lot of calibration during spill-pause)

- One could store physical and calibration triggers in different files
  - if files separated, one could decide not to keep calibrated data
  - or store only 10% of calibration events

Milestones

- Introduce calibration in DABC
  - reuse code from 'stream' framework
  - web-based display and control
  - different modes for calibrations
  - ~2 month of work

- Implement HADES event building in DABC
  - support of HADAQ format in dabc::bnet
  - run control via DABC
  - different storages for physic and calibration triggers
  - integration with EPICS
  - ~2 month of work

Conclusion

- Code exists in ‘stream’ framework and used for more than 1 year
- Several month to integrate in DABC
- Can be directly tested in laboratory and test-beams setups
- Enough time to integrate in HADES DAQ