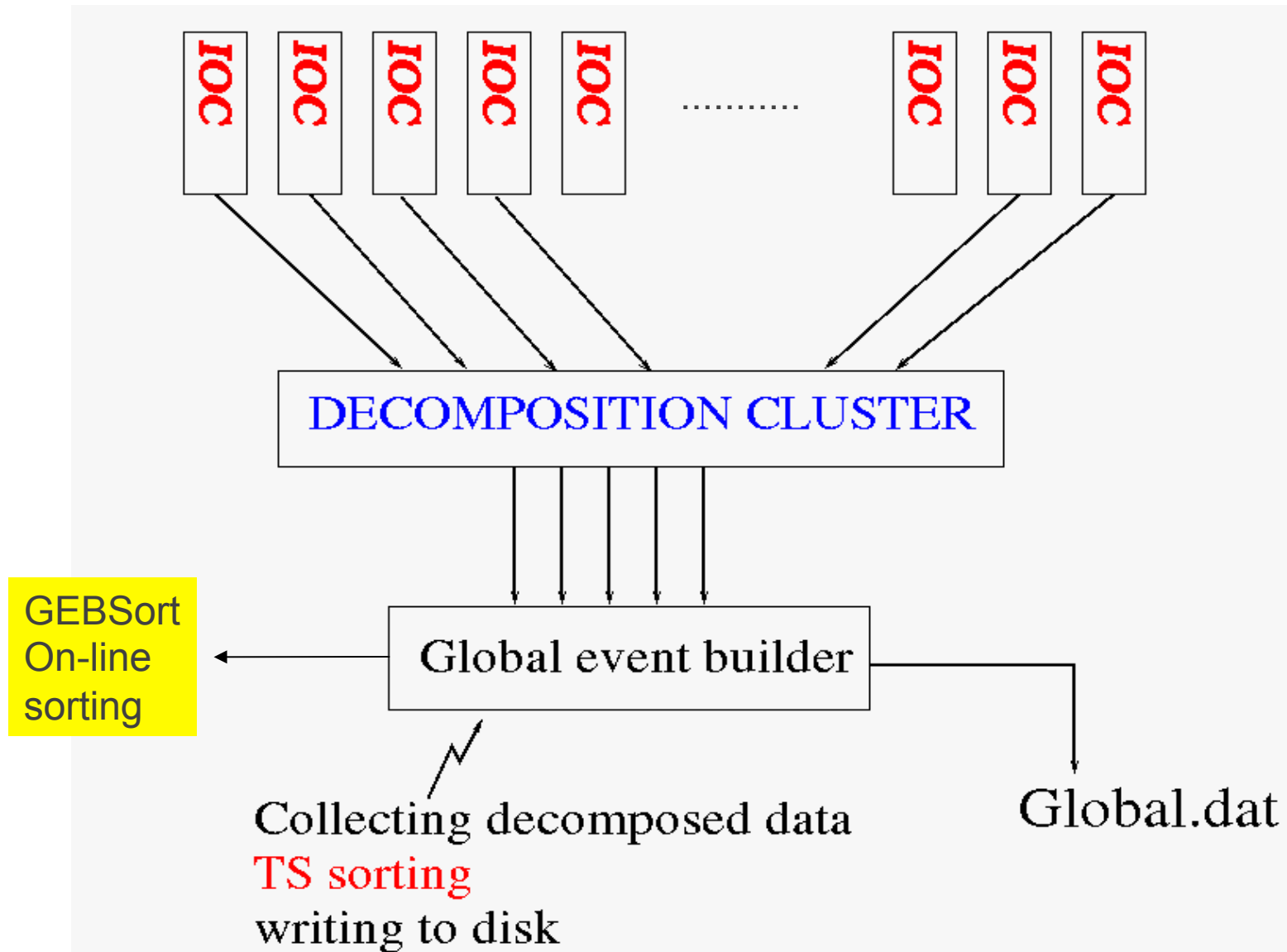


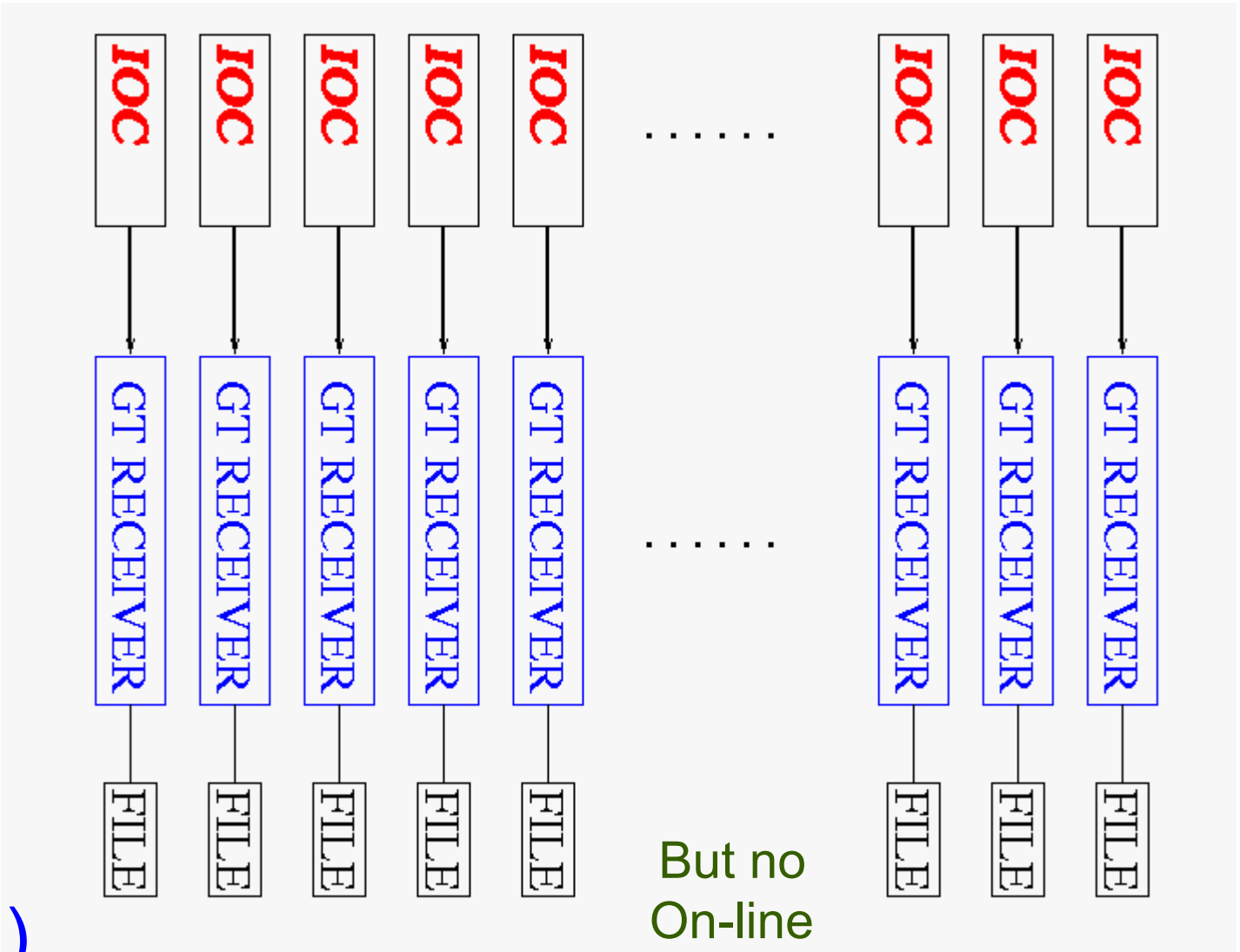
Merging and time ordering data from DGS

The GRETINA merging (simplified TL view)



GEB is a bottle neck!

The DGS data collection scheme

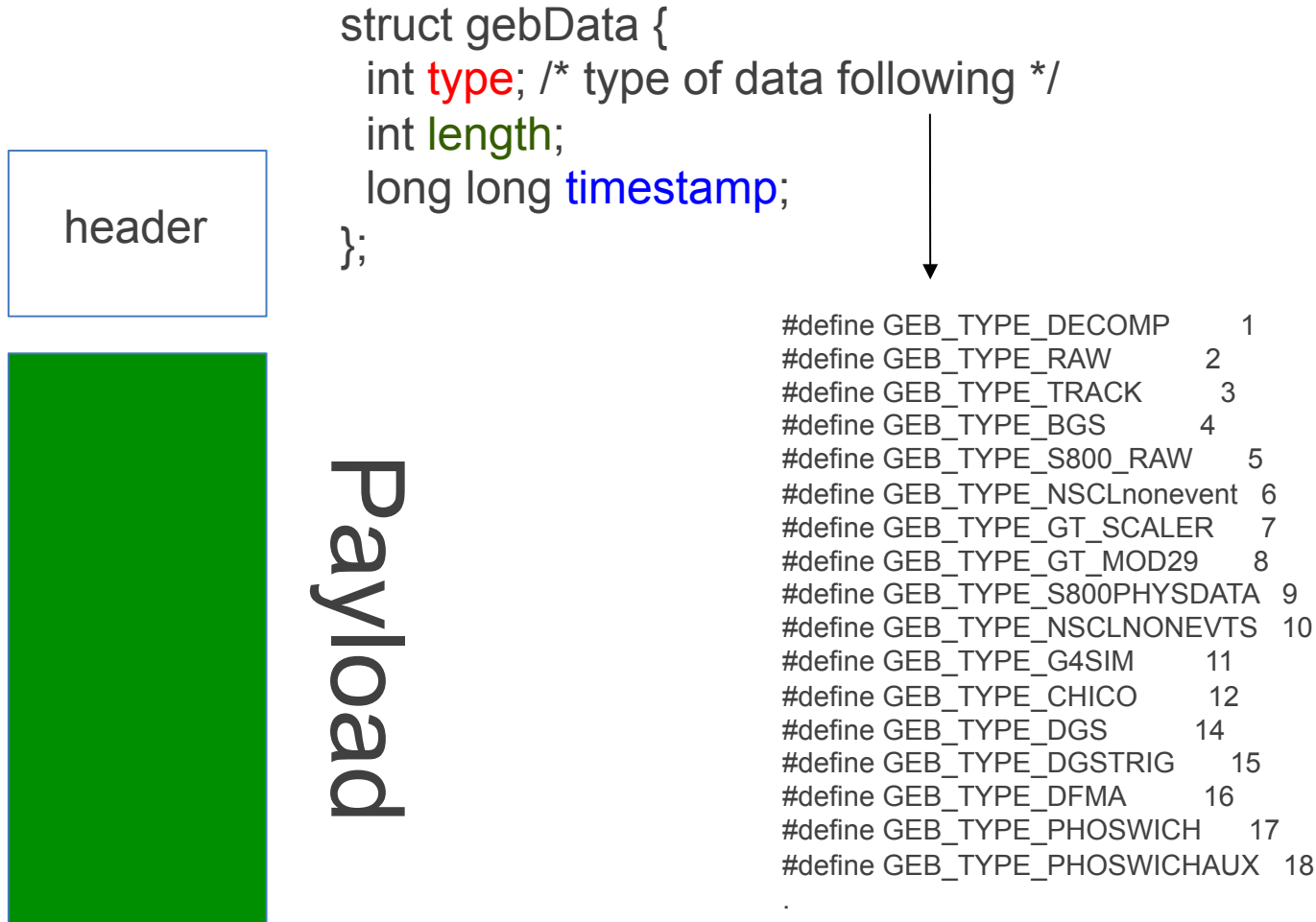


No
bottle
necks :)

But no
On-line
Sorting :(



Always use GT Header/Payload scheme also for any AUX detector systems

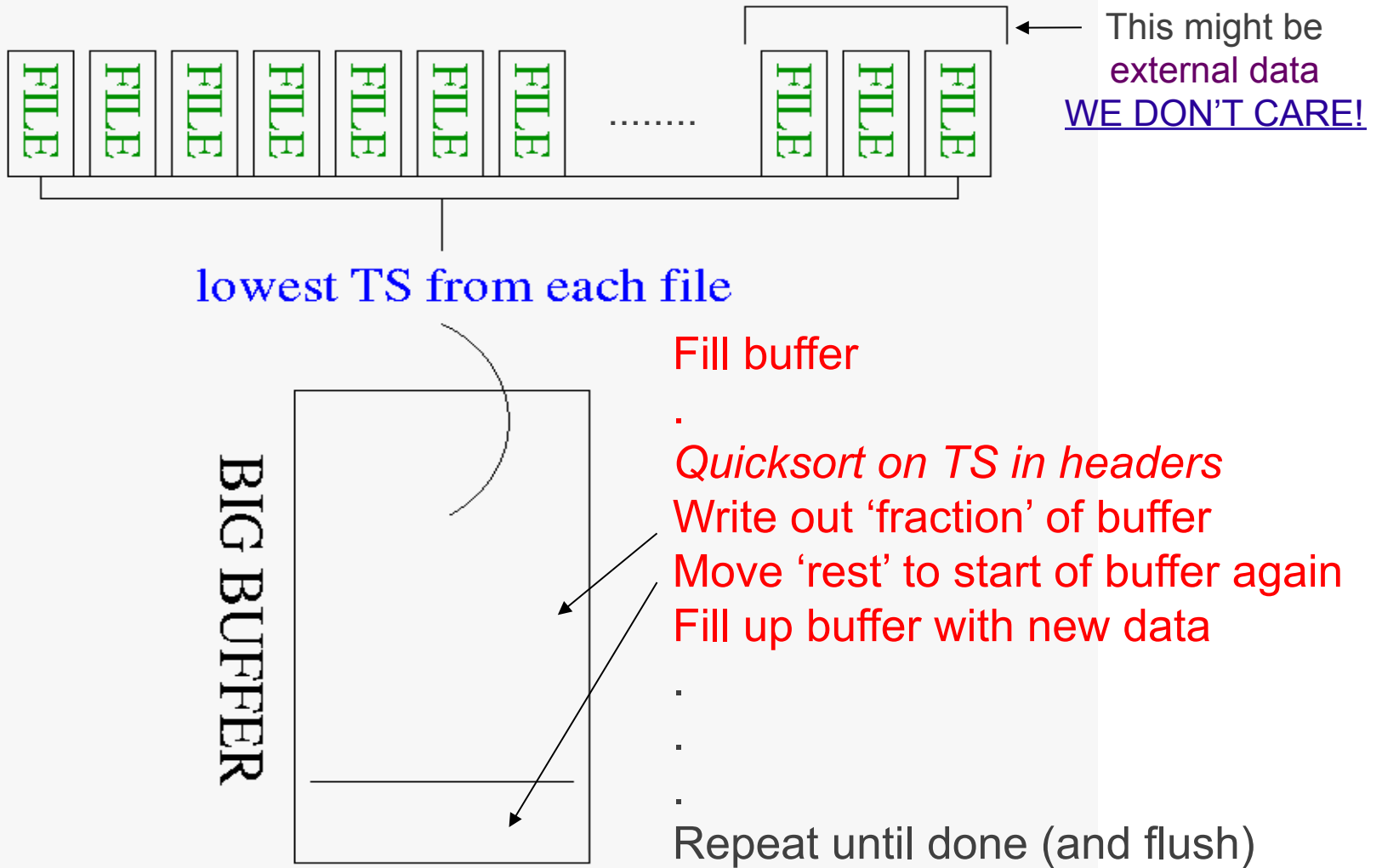


Unique numbers!
Header rules



GEBMerge

(universal, no need to know payload content)



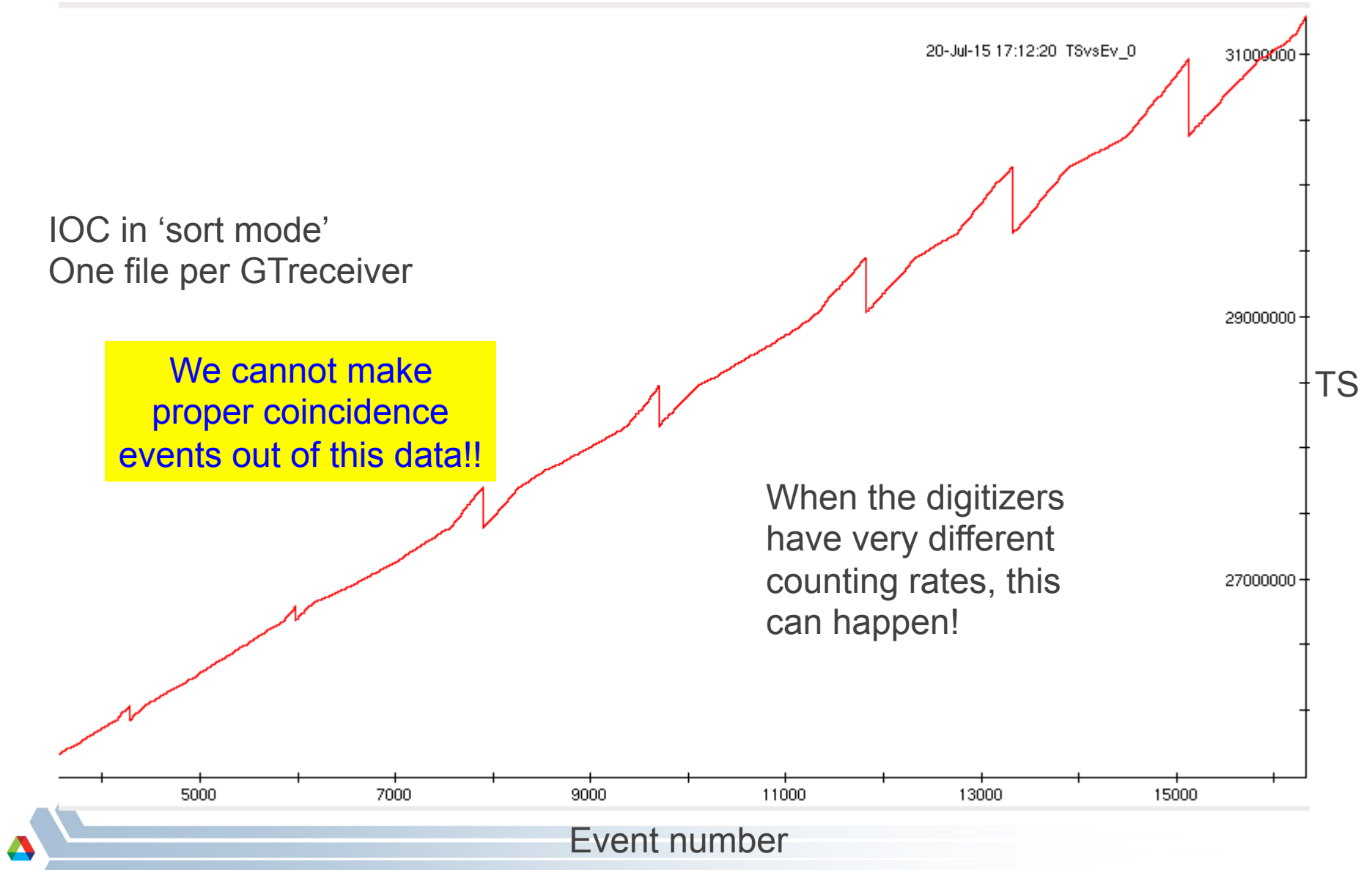
Why sort the big buffer?

Examples of TS vs event number

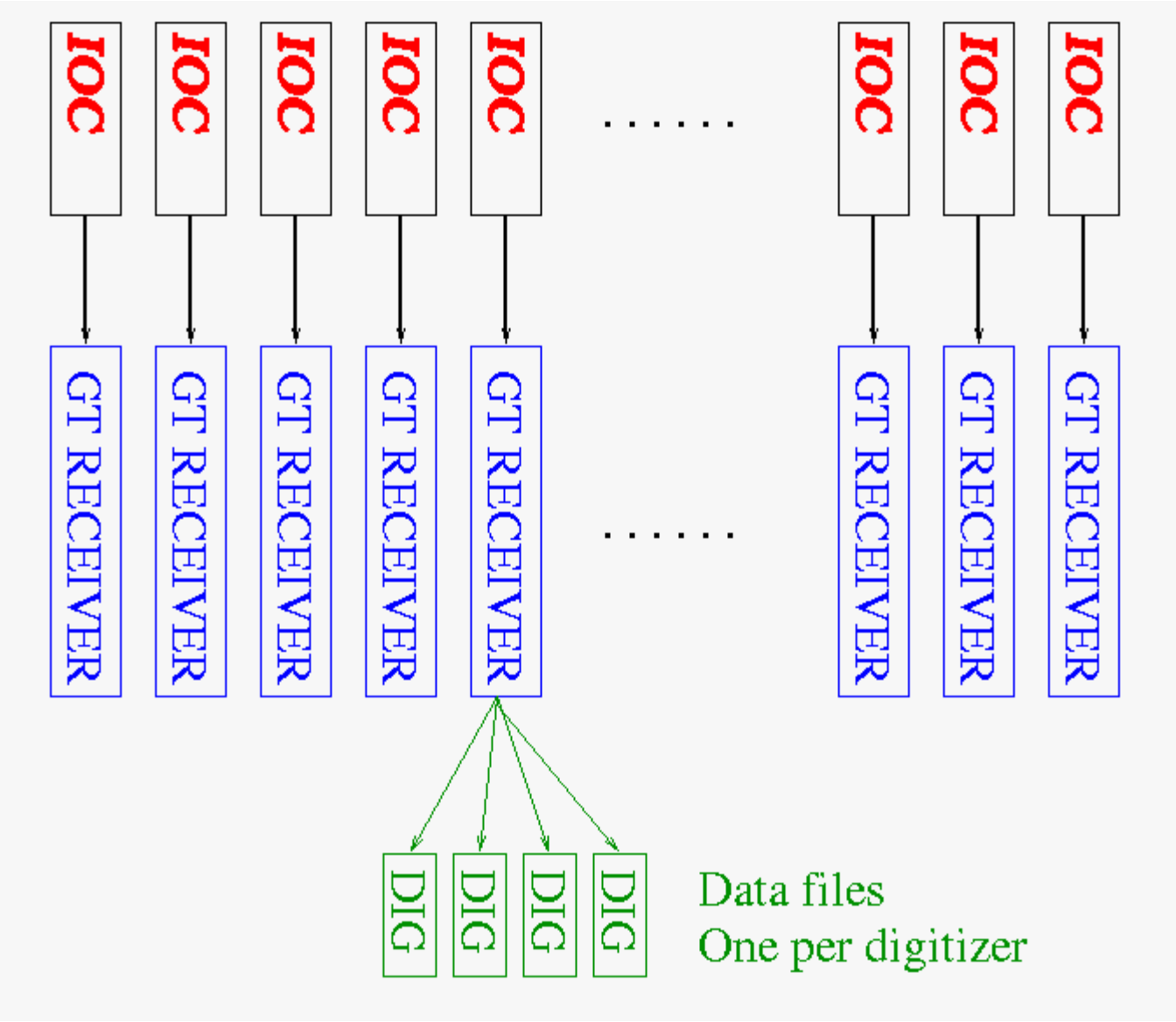
IOC in 'sort mode'
One file per GTreceiver

We cannot make
proper coincidence
events out of this data!!

When the digitizers
have very different
counting rates, this
can happen!

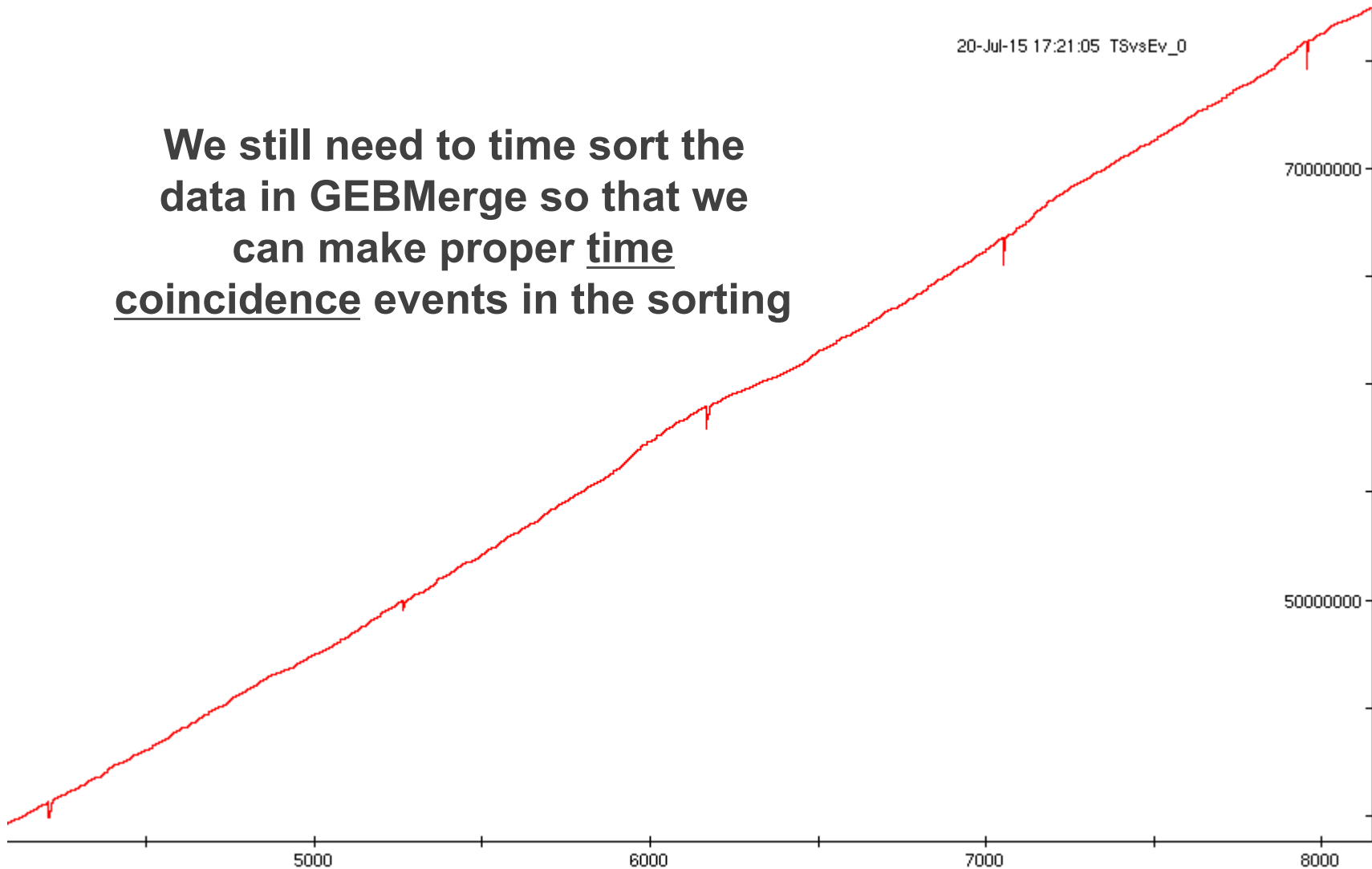


First solution: Split data according to digitizer



It is much better, but there are still glitches

We still need to time sort the data in GEBMerge so that we can make proper time coincidence events in the sorting



Time travel hurts...

- Sometimes we have TS glitches in DGS+AUX
 - **GEBMerge can hang on that**
 - Especially for a TS that jump forward because **we have to wait for the 'current' timestamps to catch up before we get at the data behind that timestamp**
 - We detect and recover...
-
- If data comes out 'late' in the datastream we need to have a buffer that is big enough to handle that, i.e., that we do not write out data to disk before we have collected and time sorted all the data that came out late
 - That happens more in DGS: channels can have very **uneven countrates**. The ones with low rates come out of the DAQ late



TS ordering in the IOC

- At the moment the IOC usually orders the data from the four (two in DGS) Digitizers before the data are sent out
-That uses a lot of IOC CPU cycles
- **We want to move that task to the receivers where we have more CPU power and we can distribute it over many Linux boxes (GTRceiver5)**



Conclusions and future

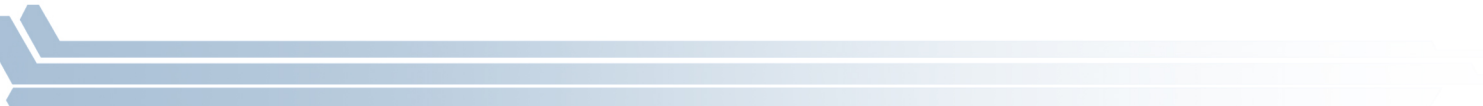
- We had much 'fun' learning to handling data based on time stamps in the beginning. There was a learning curve.
- Now we run routinely without serious problems
- Because we adapted the **GT Header/Payload scheme**, we can easily handle external data: **From my point of view: All we need is 1) a reliable way to share time-stamps between DGS and auxiliary detector systems and 2) write data in GT Header/Payload format and we are ready to handle the data!**
- We are working on ways to force readout of digitizers with low count rates **before** the FIFO is full
- **GEBSort** can handle: GT (mode2/mode1, on-line and off-line), (AG), DGS, DFMA, S800, CHICO2, etc. **Common sorting frame** with **functions that handle the different payloads based on the header ID**. We minimize the work to add a new type of data processing

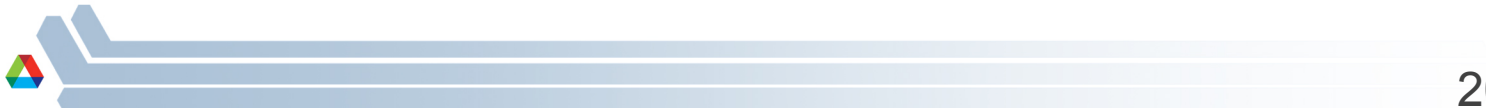




participants

- Mike Carpenter
- John Anderson
- Tim Madden
- Shaofei Zhu
- Darek Seweryniak
- Torben Lauritsen
- Phil Wilt
- Michael Oberling
- Akaa Ayangeakaa
- Helena David
- Andrew Johnson
- And others





First solution: Split data according to digitizer

