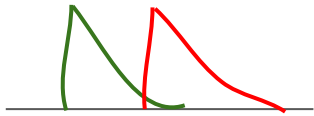


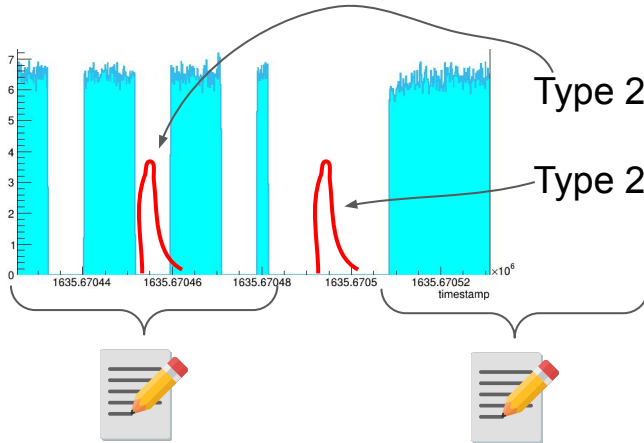
ALPACA Deadtime PSI2021

What classes of deadtime do we have in ALPACA?

Deadtime: Fraction of time the DAQ was not able to start event recording on possible triggers



Type 1) Intrace pileup; DAQ window blocked due to previous trigger



Type 2a) Deadtime within a file (“bank swap”)

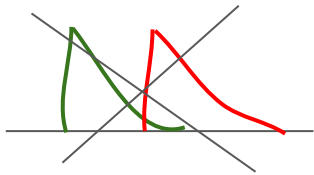
Type 2b) Deadtime between files (“file swap”)



NOT deadtime: DAQ paused

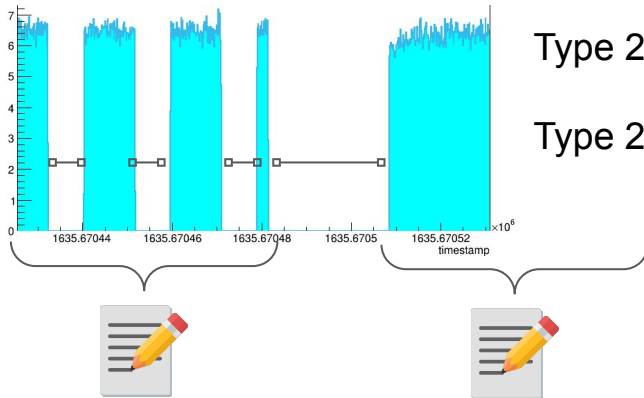
1st approach: "Gap analysis"

Check **gaps** between **consecutive timestamps**; report every gap > 100 ms



Type 1) Intrace pileup; DAQ window blocked due to previous trigger

invisible in gap analysis



Type 2a) Deadtime within a file ("bank swap")



Type 2b) Deadtime between files ("file swap")



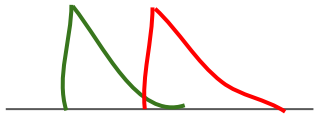
Regard any gap > 60 s as DAQ paused



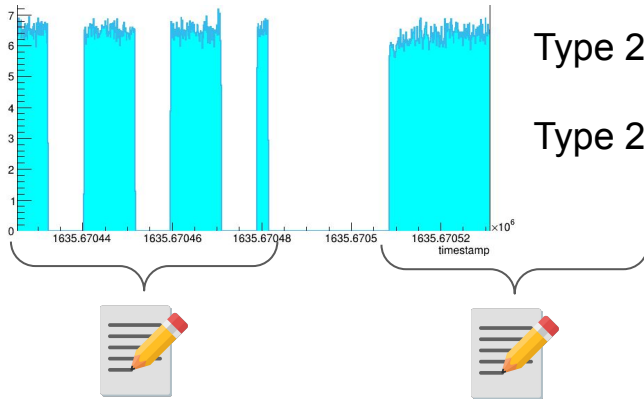
NOT deadtime: DAQ paused

2nd approach: “Test pulser analysis”

Count **number of test pulser events** recorded; we know that it should be **30 Hz** if no deadtime



Type 1) Intrace pileup; DAQ window blocked due to previous trigger



Type 2a) Deadtime within a file (“bank swap”)



Type 2b) Deadtime between files (“file swap”)

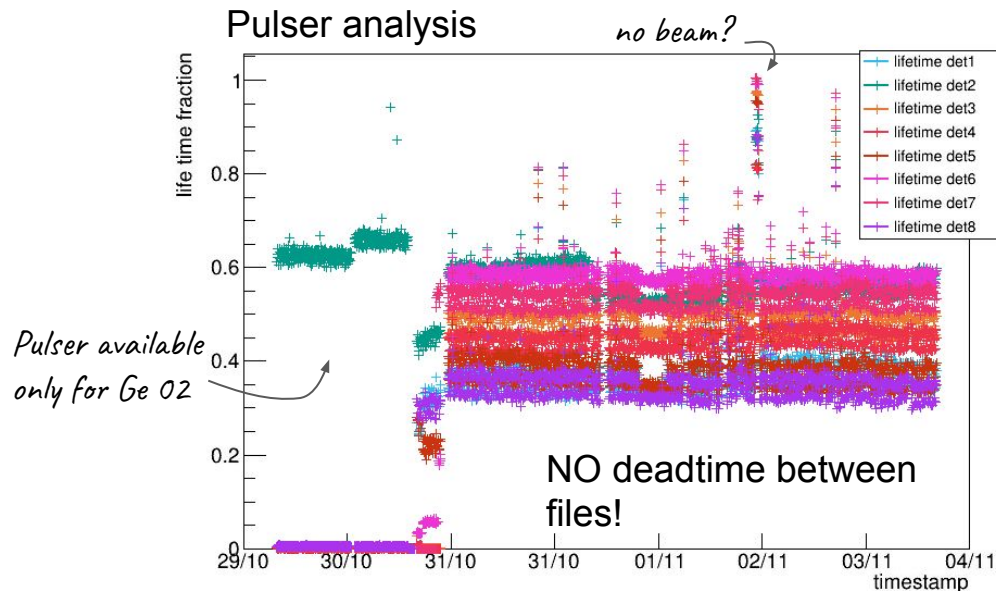
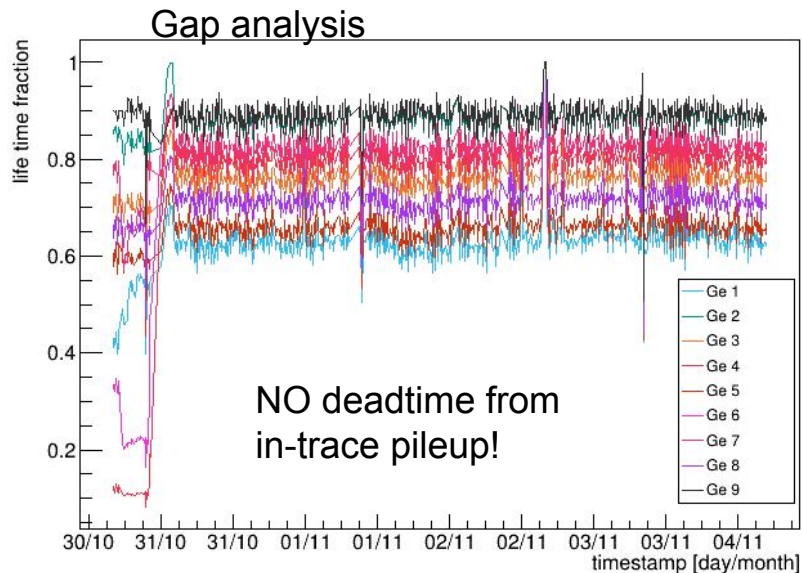
WIP:
extend test pulser
analysis also
between files



NOT deadtime: DAQ paused

Comparison: lifetime from gaps vs lifetime from pulser

Se⁷⁶

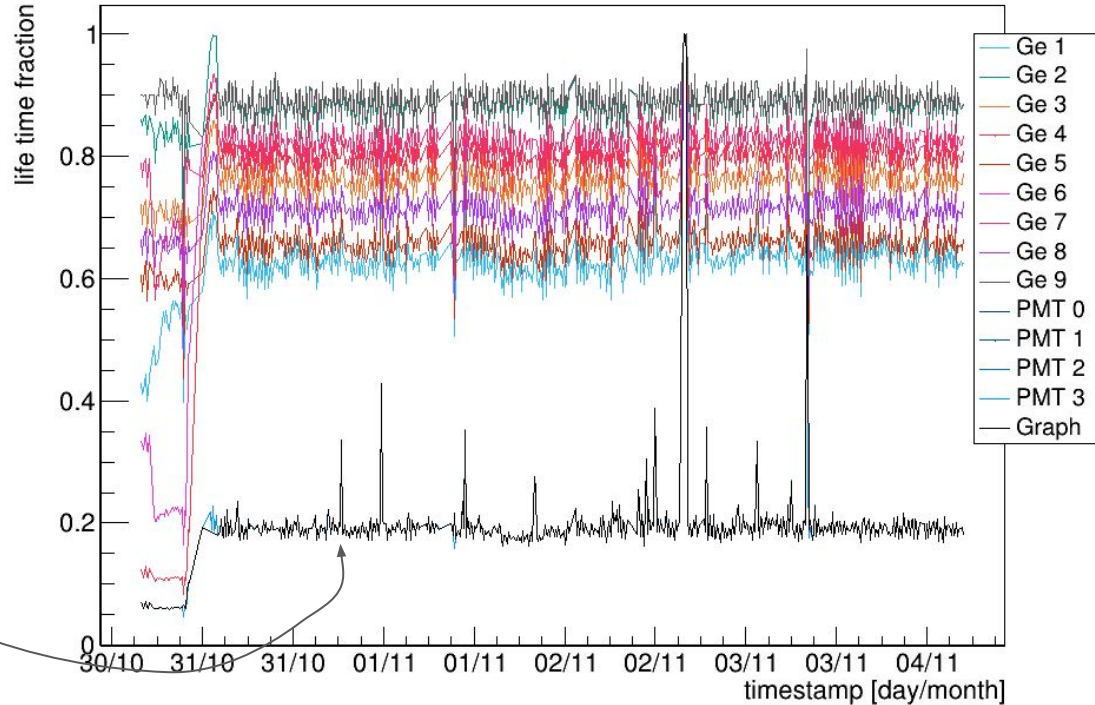


Lifetime from pulser much lower →
presumably due to in-trace pileup:
need to finish cross-check of methods!

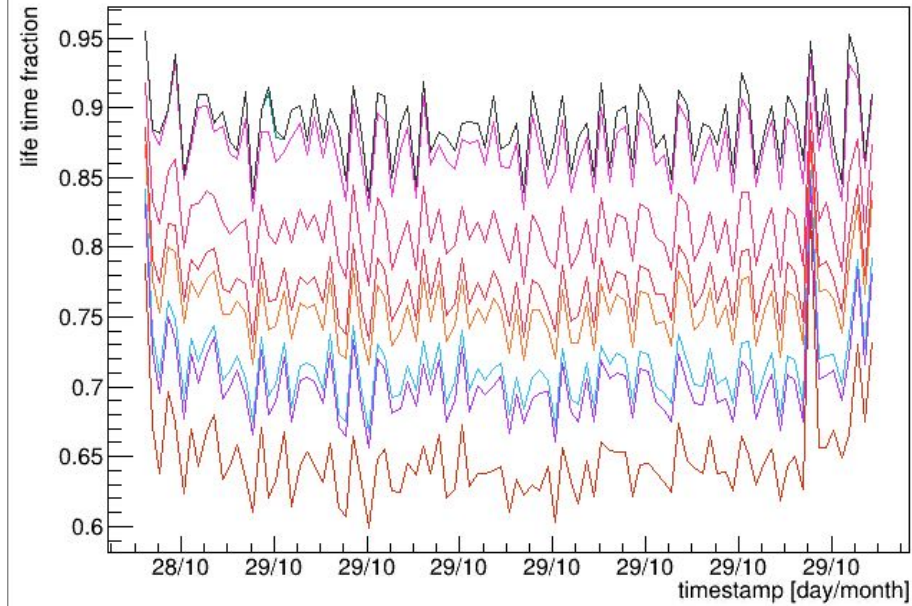
Se-76 gap analysis

Channel	Runtime [s]	lost time [s]	loss fraction [%]
Ge 1	321661	138058	42.9204
Ge 2	321706	42107	13.0887
Ge 3	321685	88249.1	27.4334
Ge 4	321678	87887.9	27.3217
Ge 5	321668	125583	39.0412
Ge 6	321692	53441.9	16.6128
Ge 7	321697	65591.3	20.3892
Ge 8	321678	105098	32.6719
Ge 9	321707	39718.6	12.3462
PMT 0	321580	298020	92.6737
PMT 1	321580	298024	92.6751
PMT 2	321580	298026	92.6755
PMT 3	321580	298020	92.6736

life time fraction of complete system driven by PMTs



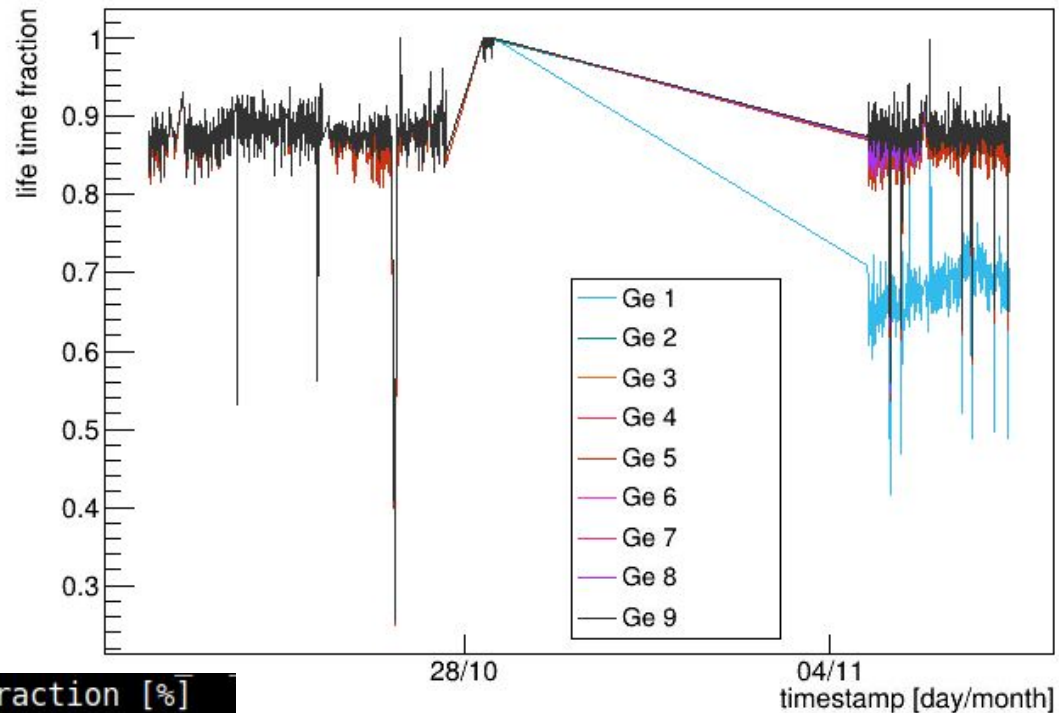
Se-nat



Channel	Runtime [s]	lost time [s]	loss fraction [%]
Ge 1	56821.9	16196.4	28.5037
Ge 2	56833	6439.75	11.331
Ge 3	56824.4	13760.6	24.216
Ge 4	56825.4	12746.6	22.4313
Ge 5	56818.3	20021.4	35.2375
Ge 6	56831.8	7176.51	12.6276
Ge 7	56827.6	10476.3	18.4352
Ge 8	56821	16885.3	29.7166
Ge 9	56833	6434.57	11.3219

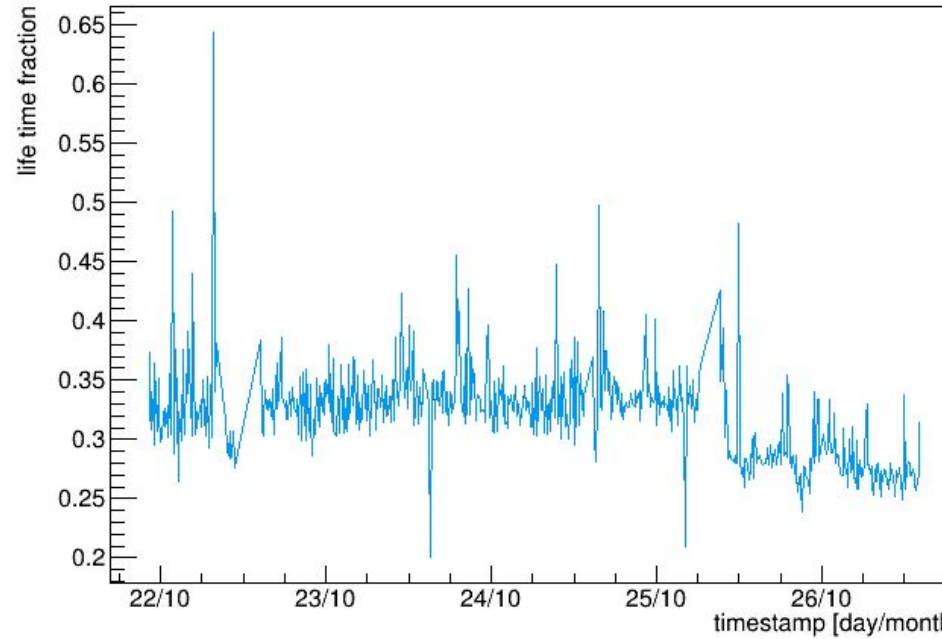
(PMT pending)

Ba-136 Ge



Channel	Runtime [s]	lost time [s]	loss fraction [%]
Ge 1	681492	123941	18.1867
Ge 2	681498	79995.7	11.7382
Ge 3	681498	79974.7	11.7351
Ge 4	681498	79974.7	11.7351
Ge 5	681497	87162.4	12.7898
Ge 6	681498	79978	11.7356
Ge 7	681498	79975.5	11.7353
Ge 8	681498	81350.5	11.937
Ge 9	681498	79989.9	11.7374

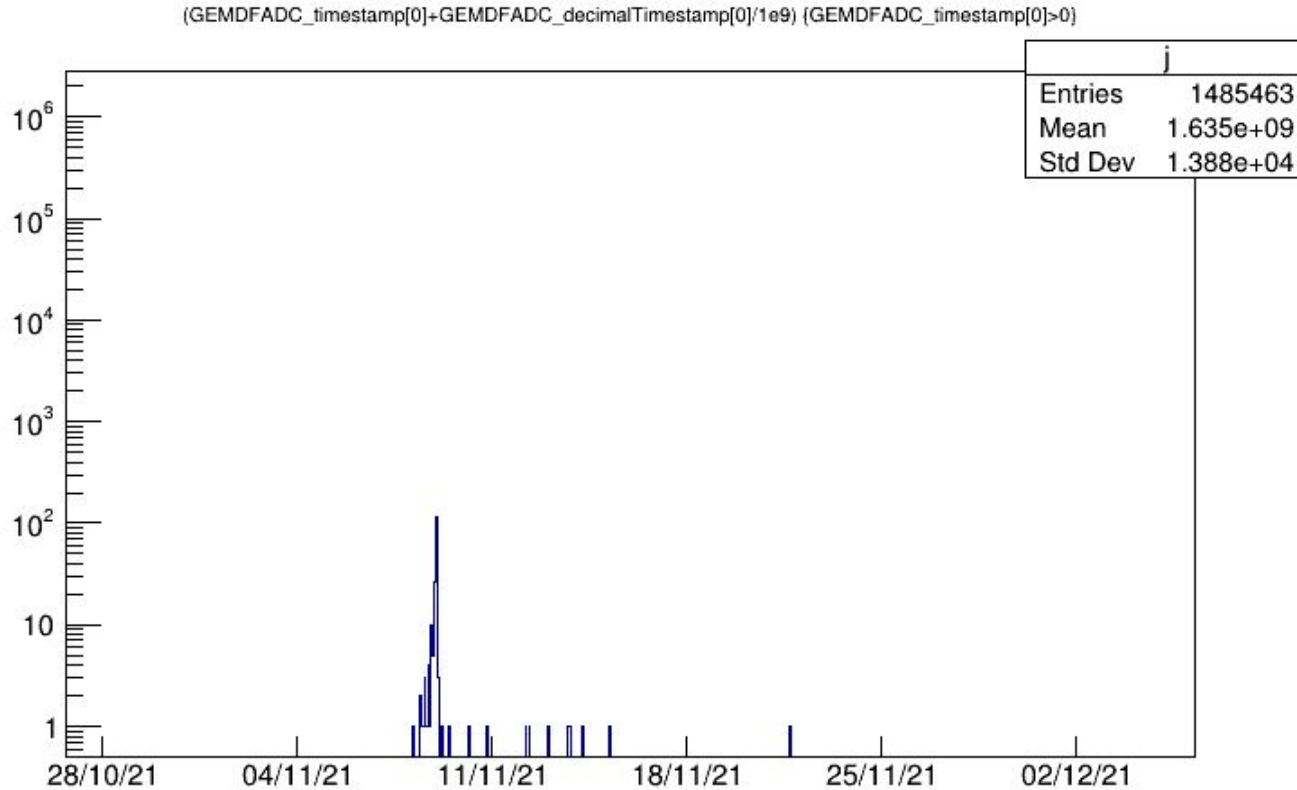
Ba-136 PMT



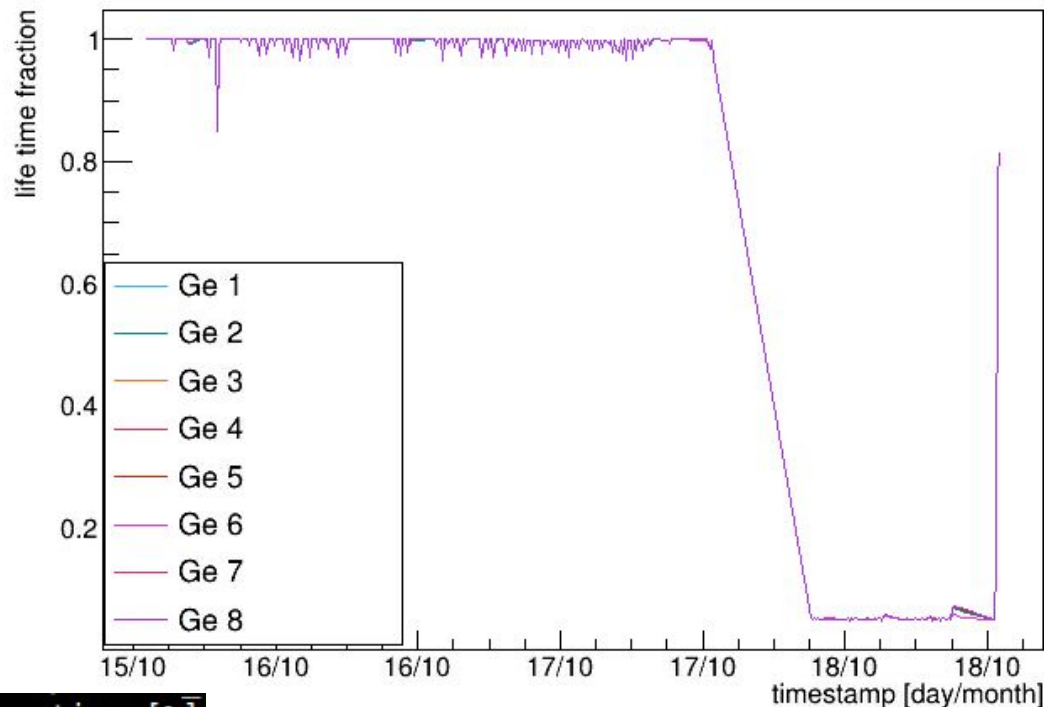
BUG! -> MESSEUP UP TIMESTAMPS

Channel	Runtime [s]	lost time [s]	loss fraction [%]
PMT 0	-4.47383e+06	460977	-10.3038
PMT 1	-5.05044e+06	462843	-9.16441
PMT 2	-6.31792e+06	461060	-7.29765
PMT 3	-3.40227e+06	460978	-13.5491

Intermezzo: in ba136-05 timestamps are messed up

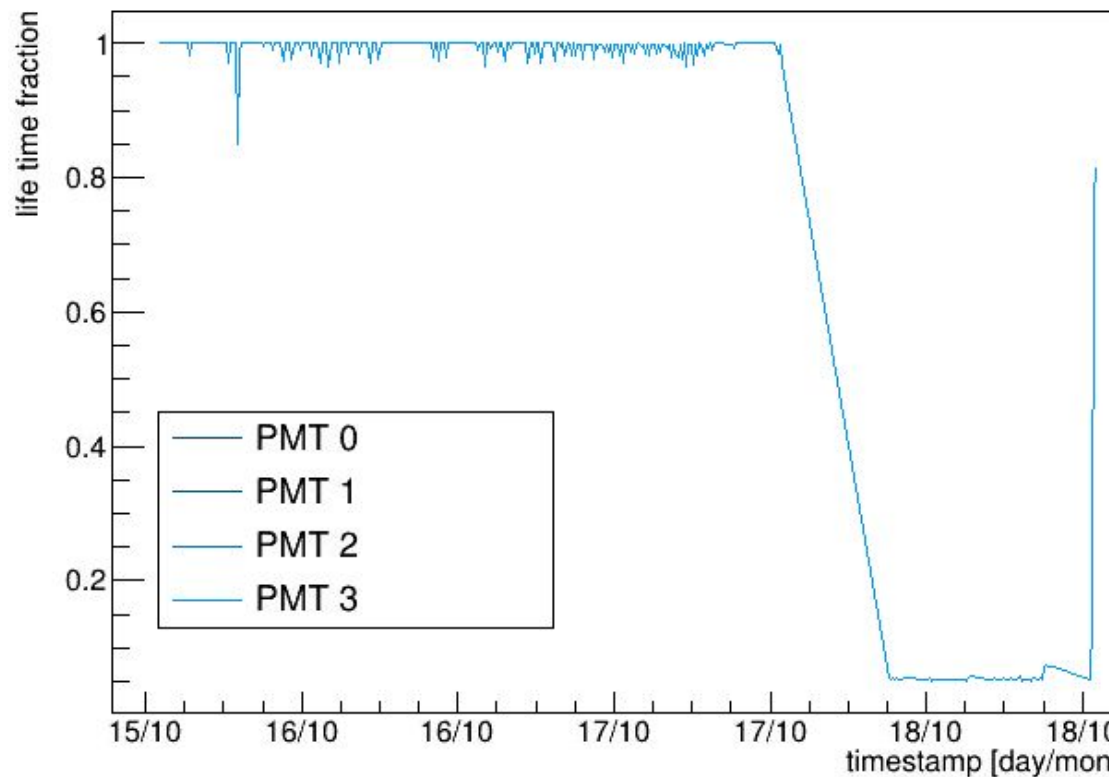


Ba-nat Ge



Channel	Runtime [s]	lost time [s]	loss fraction [%]
Ge 1	172583	36034.9	20.8797
Ge 2	172583	36072.4	20.9015
Ge 3	172583	36030.9	20.8774
Ge 4	172583	36035.2	20.8799
Ge 5	172583	36037.9	20.8815
Ge 6	172583	36102.2	20.9187
Ge 7	172583	36030.4	20.8771
Ge 8	172583	36032.1	20.8781

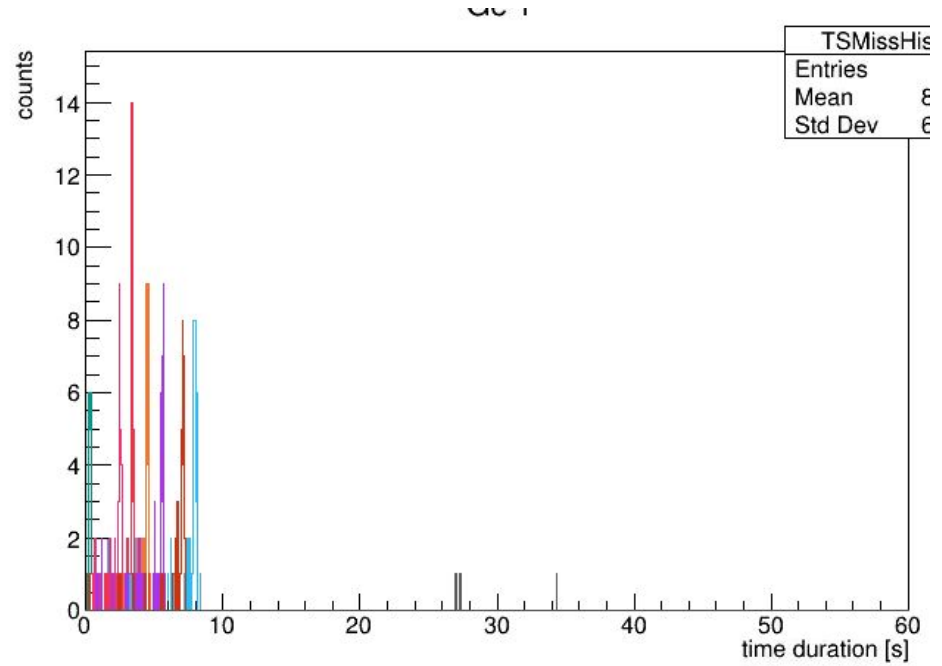
Ba-nat PMT



Channel	Runtime [s]	lost time [s]	loss fraction [%]
PMT 0	172583	36029.5	20.8766
PMT 1	172583	36029.5	20.8766
PMT 2	172583	36029.5	20.8766
PMT 3	172583	36029.5	20.8766

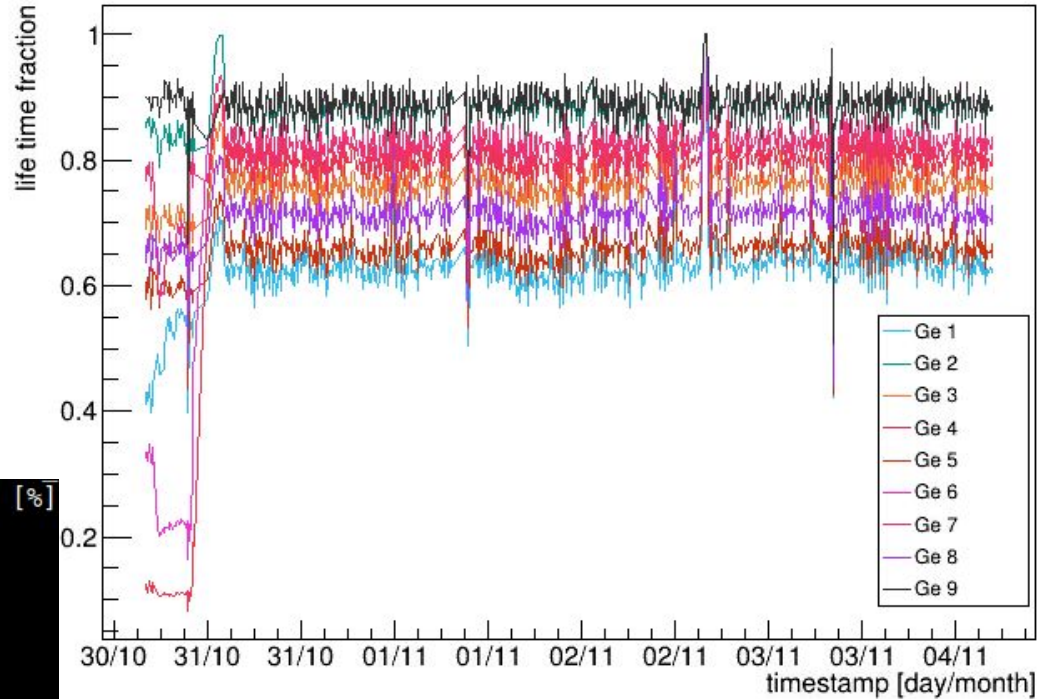
Backup

Size of gaps in regular data
taking much smaller than 60 s

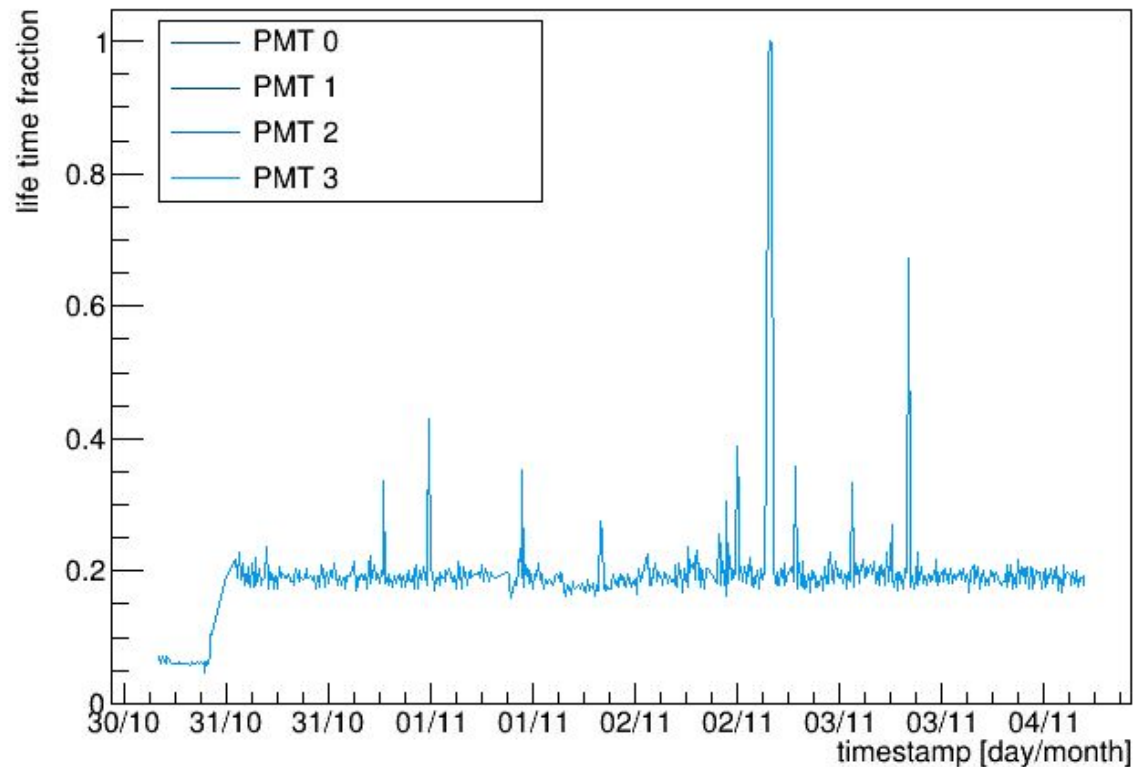


Se-76 Ge

Channel	Runtime [s]	lost time [s]	loss fraction [%]
Ge 1	321661	138058	42.9204
Ge 2	321706	42107	13.0887
Ge 3	321685	88249.1	27.4334
Ge 4	321678	87887.9	27.3217
Ge 5	321668	125583	39.0412
Ge 6	321692	53441.9	16.6128
Ge 7	321697	65591.3	20.3892
Ge 8	321678	105098	32.6719
Ge 9	321707	39718.6	12.3462

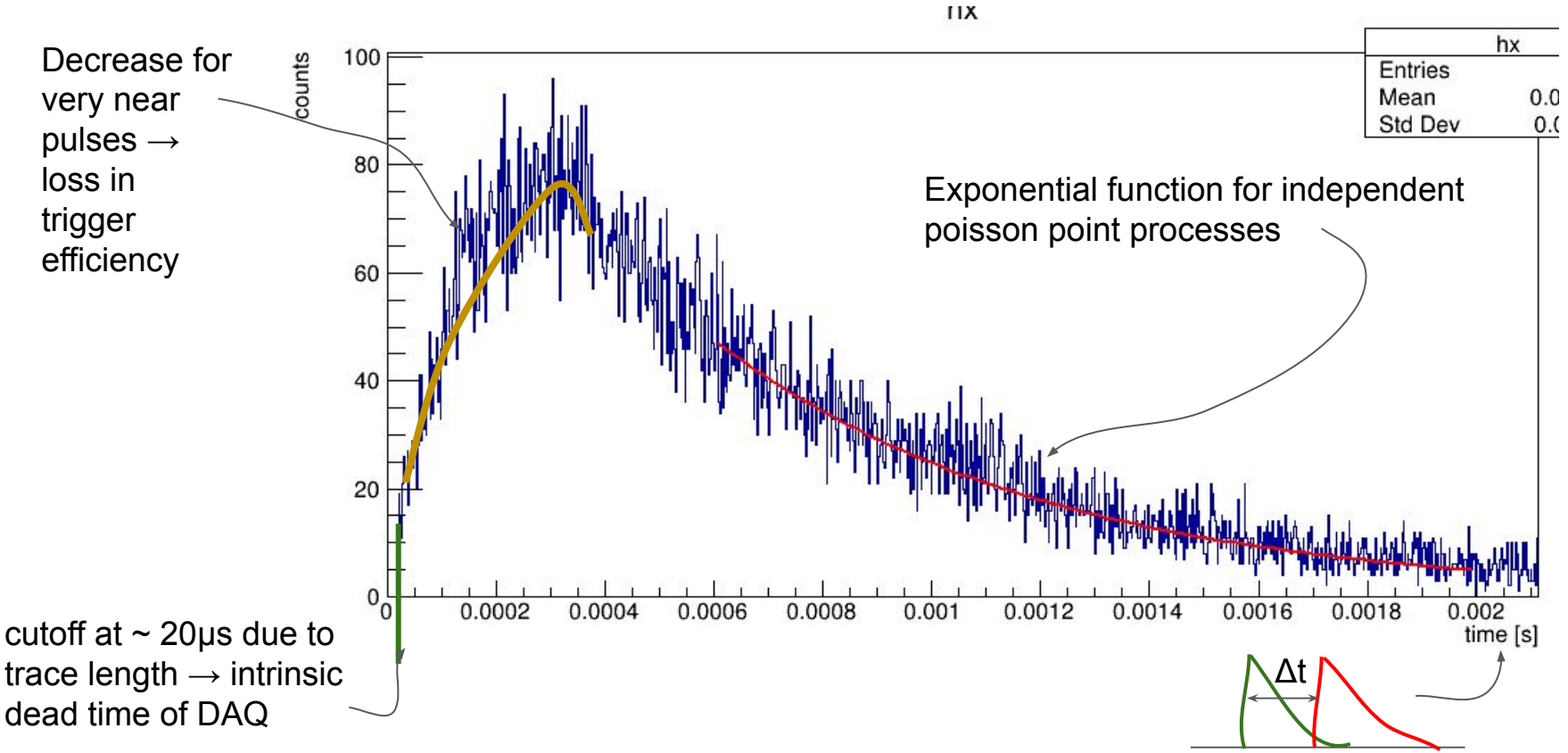


Se-76 PMT



Channel	Runtime [s]	lost time [s]	loss fraction [%]
PMT 0	321580	298020	92.6737
PMT 1	321580	298024	92.6751
PMT 2	321580	298026	92.6755
PMT 3	321580	298020	92.6736

Deadtime from intrace pileup



Intrace pileup

Efficiency to detect 2nd pulse

Even for very close pulses, no dependence on energy of 2nd visible → looks like we have **no energy-dependent trigger efficiency**

