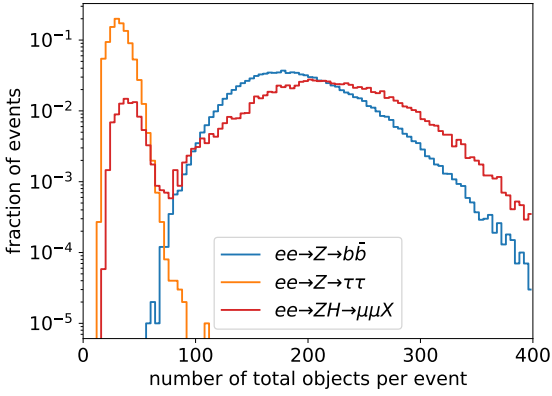
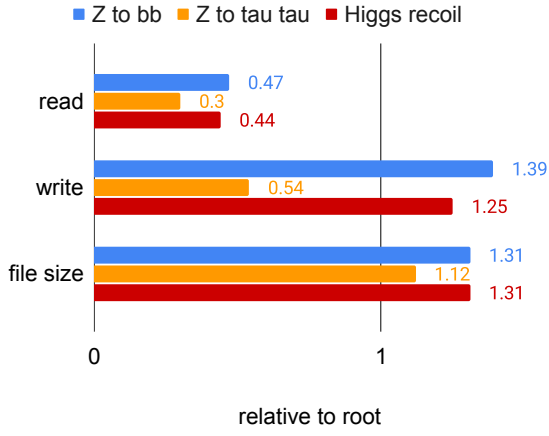
podio benchmarks (for vCHEP)

- Comparing ROOT and SIO backend for three different physics use cases
  - $ee \to Z \to b\bar{b}$ @ FCC-ee, $\sqrt{s} = 91$ GeV (100k events)
  - $ee \to Z \to \tau\tau$ @ FCC-ee, $\sqrt{s} = 91$ GeV (100k events)
  - $ee \to ZH \to \mu\mu X$ (Higgs recoil) @ ILD, $\sqrt{s} = 250$ GeV (~ 17 k events)
- "Generating" events and simulating detector response with `k4SimDelphes` tool for write benchmarks
- Read benchmark is a standalone executable that simply "touches" all present collections at least once
  - Somewhat similar to framework I/O access patterns (?)
- Using **default settings for both backends**
  - Not the fairest comparison between ROOT and SIO, since compression algorithms and levels can be different
  - But no easy way for podio users to change these defaults (without re-compiling podio)
  - (Nearly) endless opportunities for tuning this, especially for ROOT
- Currently using ROOT v6.20/04 (v6.22/06 with some unresolved performance degradation) and SIO v0.0.4

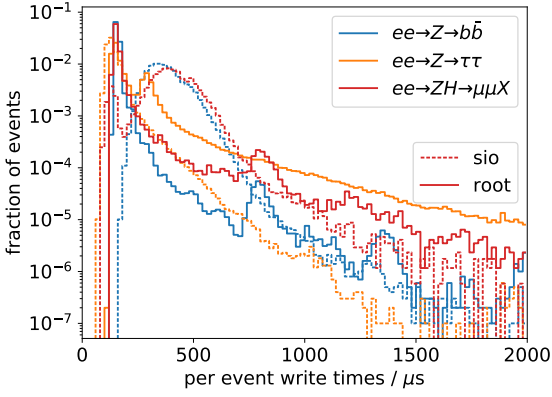
Event contents and overview results

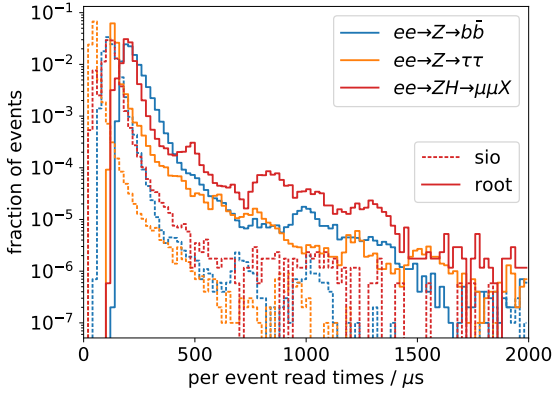




- Cover a fairly wide range of event contents
- Overall results to be revisited, $ee \to Z \to \tau\tau$ benchmarks seem to be off

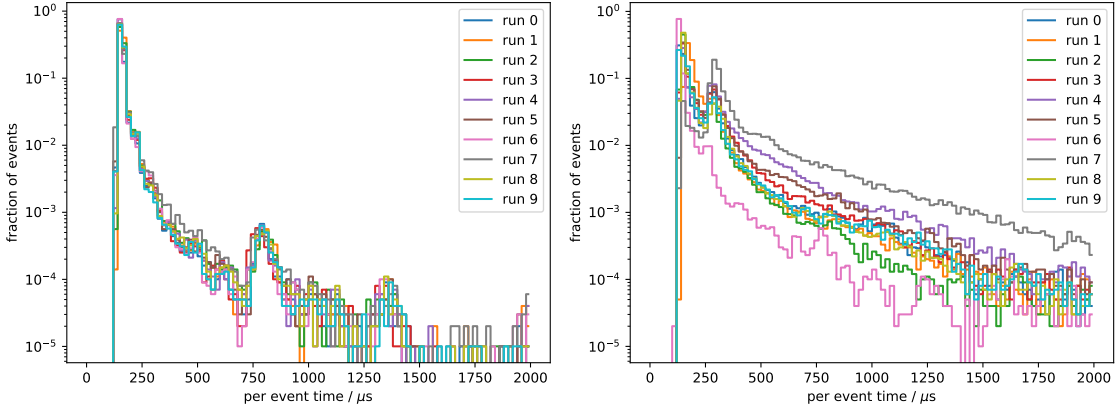
Per event I/O times





- Why are $ee \to Z \to \tau\tau$ write times longer than the others for ROOT even though there is on average less in the event?
- SIO times show similar structure as event contents (expected)
- Better format for these plots?

$ee \to Z \to \tau\tau$ benchmarks to be redone?



- Much more variation between runs for $ee \to Z \to \tau\tau$ (right) compared to $ee \to Z \to b\bar{b}$ (left)
- But still $\tau\tau$ has overall higher average times than $b\bar{b}$
- Run benchmarks again over night, see if this could be interference from other processes