



Monitoring Muon Data Online

Dennis Shpakov

Northeastern University

June 17, 2003

Muon Parallel Session

DØ Workshop at Beaune

Overview

1. Existing Tools
2. Limitations and Missing Pieces
3. Possible Ways of Improvement

This talk is about monitoring *quality* of the *data* coming out of the detector, not data taking efficiency or detector stability.

What Is Available

1. *Real time* quality checks on the module headers: synchronization errors or hardware failures during data transmission from the platform to MCH3. Such events are either flagged in the headers or never make it past the readout crate.
2. *Quasi-real time* checks:
 - 1553 alarms on HV trips and some other FE conditions known to be fatal for data taking.
 - Conditions made available to 13xmonitor (a. k. a. DAQAI): trigger term rates, *etc.*
3. *Asynchronous* checks: examine and global monitor.

What's Missing

1. Luminosity database feedback from SES (see Tom's talk).
2. Availability of information about non-standard running conditions (disabled FE's, HV trips, *etc.*) *offline*.
3. Monitoring checksums for PDT's (and other subsystems?). A recent work on debugging a PDT hardware problem showed that checksum errors are back at a rate as bad as ~ 0.0001 per FE, which translates into hundreds of bad events in a typical 4-hour long run.
4. Examine does not serve up to its purpose: it takes an experienced shifter with good pattern recognition skills to spot problems by looking at the examine histograms.

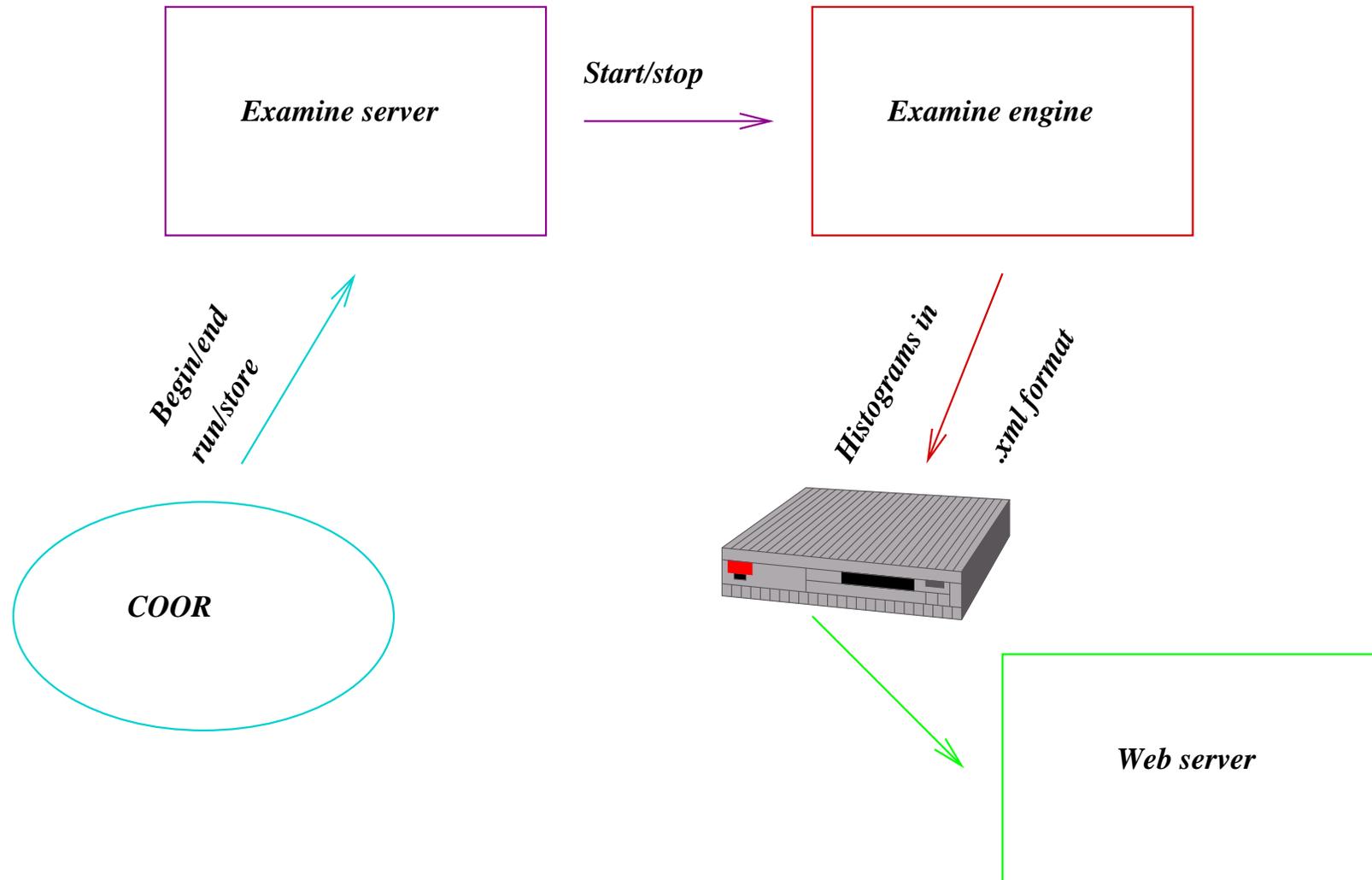
Checksums

- For the checksum errors that are not due to wrong header fields there is no way to catch them without verifying the checksums for every event.
- m68k's are too slow to do that. PowerPC's may or may not handle it (need a rate test).
- If enabling checksum calculation raises the busy fraction in the PPC's, there are two alternatives:
 1. Take end of store special runs at lower rates with checksums on. Not a real solution — only gives an idea of what FE's are on average worse than others.
 2. Have the L3 muon filter do that. Needs a data format change (hopefully backward-compatible).

Examine Wish List

- Add reference distributions and a possibility to generate “snapshots” and load them as reference distributions later.
- Have the examine program calculate matching criteria (χ^2 , maximum bin difference, Kolmogorov probability) between the current state and the reference distributions. Alarm on large discrepancies through DAQAI or SES.
- Have a server process automatically initialize a new set of examine distributions at the start of a new run and store and save them and attach to e-log at the end.
- Make the examine output available through a web interface.

Examine Wish List (*continued*)



Conclusions

- We have mechanisms in place to respond to some conditions fatal to data quality online but not to all known ones.
- Monitoring of the checksum errors will be addressed in the near future.
- Examine can and should be turned into a useful tool. Cannot commit myself 100% to implementing it but willing to take a part.