CALIBRATION

SHUTTLE(1), GRP, PROCEDURES(2)

ALICE Offline Week, October '09

Chiara Zampolli

with the help of: R. Brun(2), C. Cheshkov(2), A. Gheata(2), Jan Fiete Grosse-Oetringhaus(1), P. Hristov(2), A. Morsch(2)

Outline

- **The Shuttle**
  - *DCS-Shuttle interaction*
  - *Shuttle logic wrt run processing*
  - *Problems/Issues during the last months*
  - *The Trigger Preprocessor*
- **The GRP**
  - *Current treatment of the data*
  - *Reasons for failure*
- **The Offline Calibration**
  - *The framework*
  - *State of the art*

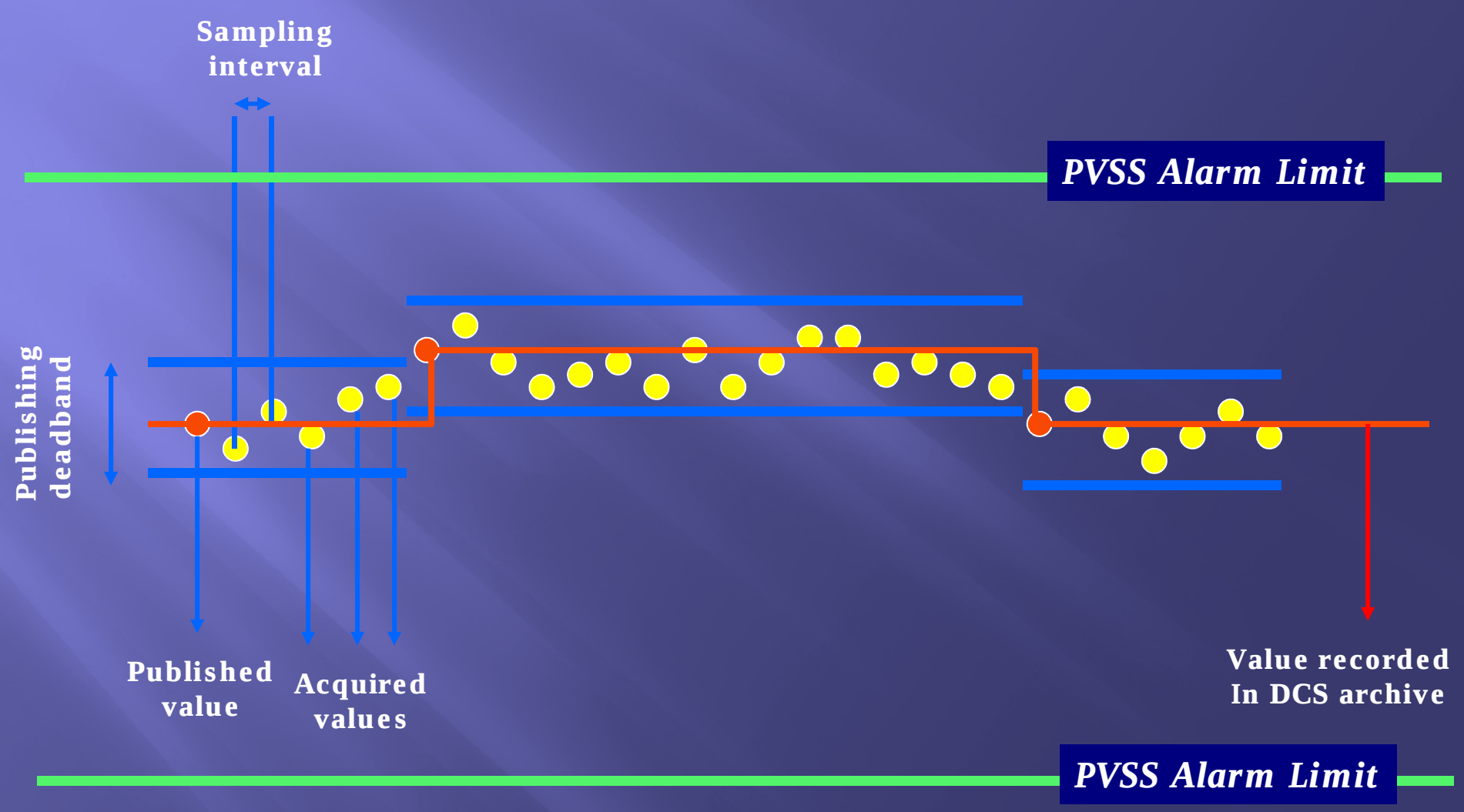
The Shuttle

DCS – SHUTTLE Interaction

- ***The Shuttle is the only way through which the detectors can access the DPs stored in the DCS archive database and have them available in the offline world***
- ***The DCS – SHUTTLE interaction strongly depends on the DCS archival procedures***
- ***The procedures used to treat/process the data as well as the usage of the subsequent output depend on the detectors themselves – the Shuttle is completely blind wrt this***

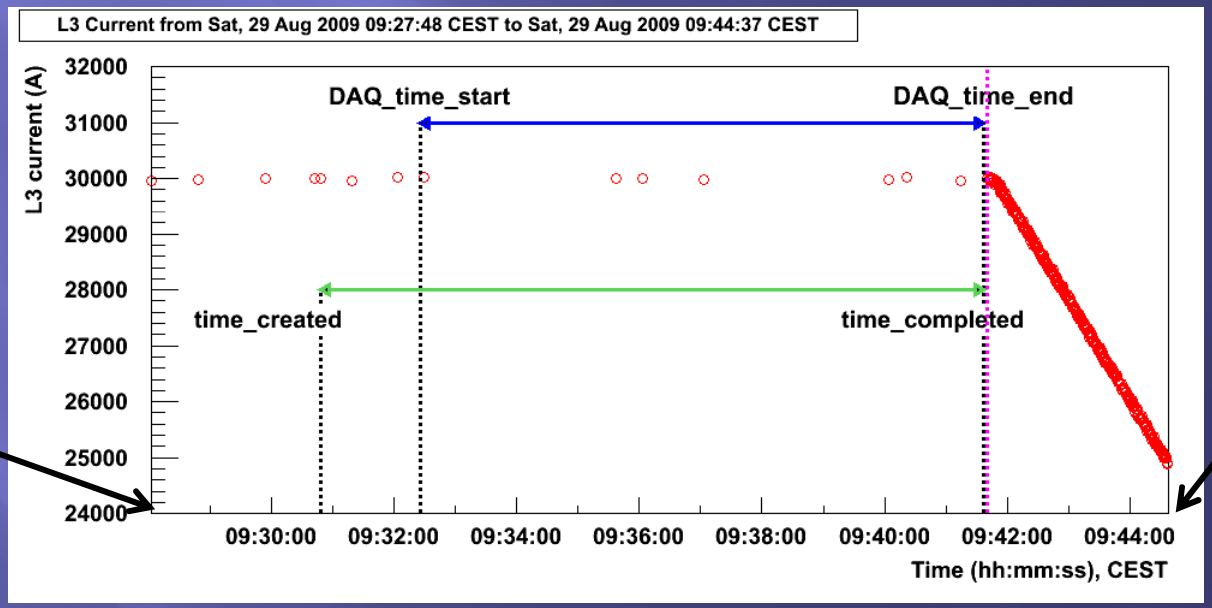
- ***Each DP stored in the DCS archive is retrieved as a pair:***
  → (value, timestamp)
- ***Each DAQ logbook field for the current run is made available to the preprocessors by the Shuttle:***
  → AliPreprocessor::GetRunParamater(const char* param)
  → ***DAQ_time_start, DAQ_time_end could be compared to the timestamp of the DP value.***

The DCS Archival



The Current Status

- *In agreement with DCS, query within:*
  [time_created – 180s, time_completed + 180s]
  → *To ensure at least two values per DP (detectors + global)*
- *Example:*



Start of the DCS query

End of the DCS query

DAQ_time_start = *SOR*

DAQ_time_end = *EOR*

time

time_created = *time at the run number creation*

time_completed = *time at the complteion of all ECS procedures*

The Aim

- *To identify methods to reduce as much as possible the time interval for the DCS retrieval, on the base of*
  - *the SOR/EOR procedures implemented in the last year by the detectors and, lately, for the global variables*
  - *DCS/ECS inter-connection*

Minutes of DCS/ECS/Offline Meeting 07.10.09

1. *In all runs started with the ECS (→ processed by the Shuttle, DCS will receive SOR and EOR signals.*
2. *With the next (wrt 07.10.09) DAQ/ECS release, the EOR will be issued only once all systems have processed the SOR.*
3. *For SOR as well as EOR a value per DP is archived in DCS, provided that the detectors have implemented the correct procedures. This is the case for all detectors at present. Both of these DPs are within the time stamps time_created and time_completed in the DAQ logbook → 2 values assured per DP per run*
4. *The Shuttle should query the values within time_created and time_completed without any additional offset. All these values are provided to the subdetectors' preprocessors.*
5. *Via the Shuttle, the time stamps DAQ_time_start and DAQ_time_end, which is the period where data was taken, can be retrieved.*
6. *The Shuttle should not query DCS earlier than 60 seconds after time_completed to allow the buffers in DCS to be flushed into the database.*

Minutes of the Meeting – II

7. ***The subdetectors' preprocessor has to decide if values outside of DAQ_time_start and DAQ_time_end are used in their procedures or not.***
   (NB: All values have time stamps.)
8. ***The subdetectors' preprocessor should catch trips (e.g. of a HV) which are allowed to occur in a run***

Further Remarks by P. Chochula to point 4:

Adding an offset would allow to recover from cases where the SOR/EOR archiving (point 3 above) fails. In such a case the subdetectors would need to decide if they use the data that is provided outside time_created and time_completed themselves.

Detectors Feedback from Current Offl. Week

ACO	???	*SPD*	OK
EMC	OK	*SDD*	OK
FMD	OK	*SSD*	OK
GRP	OK	*TOF*	OK
HMP	OK	*TPC*	OK
MCH	OK	*TRI*	OK
MTR	OK	*TRD*	???
PHS	OK	*T0*	OK
CPV	OK	*V0*	OK
PMD	Need checking	*ZDC*	OK

OK → *no problem*

??? → *waiting for confirmation*

Deadline set on Friday

The SHUTTLE Run-Processing Logic

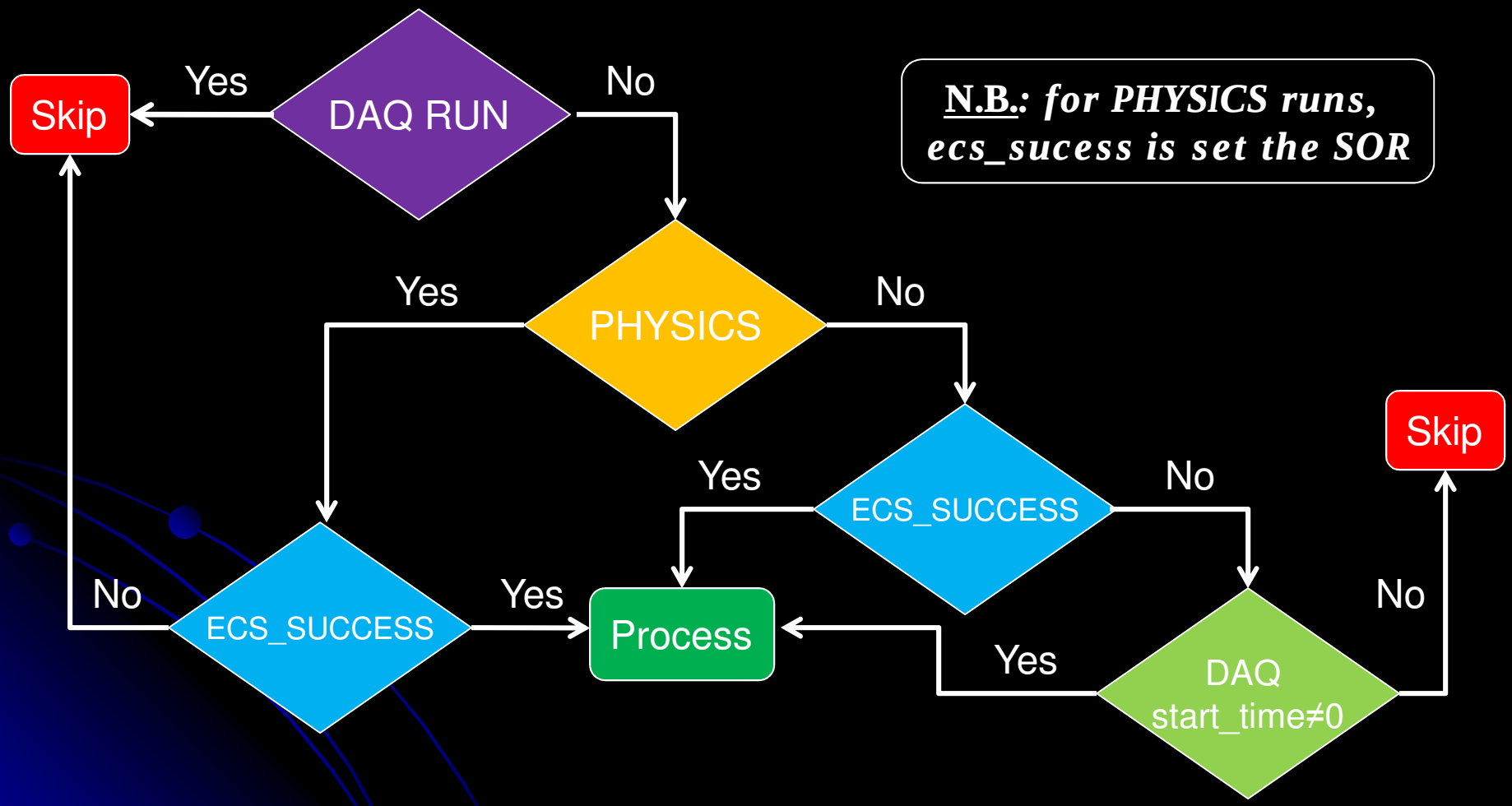
- *Currently:*
  - *if (time_created>0 && time_completed>0 && (n. of events >1 || ecs_success) →run processed*

    NB. ecs_success == kFALSE does not prevent a run from being processed, provided n.events > 1

    NB. Check on DAQ_tim_start/end not done

    NB. 120 seconds waited for DCS DPs retrieval
- *New logic discussed on 13.10.09 within ECS/DAQ/Offline*
  - *All runs started properly will be processed*
  - *Will be implemented after the Offline Week if no objections are arisen*

Shuttle – run processing and rejection logic



N.B.: *for PHYSICS runs, ecs_sucess is set the SOR*

L. Betev

Problems/Issues during the Latest Data Taking

- **DCS DPs:**
  - ***Procedure to introduce new DPs:***
    1. ***Contact Peter Chochula & Shuttle team to include the new DPs in the simulation***
    2. ***As soon as a confirmation from DCS is provided about the availability of the new DPs, contact the Shuttle team to have them included for the nightly Test Setup***
    3. ***As soon as the nightly tests succeed, and the new DPs are defined and archived at P2, contact the Shuttle team to have them included in the configuration for data taking***

→ *Please, check the aliases in $ALICE_ROOT/SHUTTLE/schema (test setup) anc $ALICE_ROOT/SHUTTLE/schema_prod (P2), and on the offline web page and contact the Shuttle team and DCS responsibles in case of mismatches*

Problems/Issues during the Latest Data Taking – II

- **DAQ/DCS/HLT DAs and FXS**
  - *New DAs versions that require corresponding changes in the preprocessors' code should be* **synchronized** *with the installation of the corresponding AliRoot tag for the Shuttle*
  - *The* **FileId** *used in the preprocessor should be the same as the one used in the DA, otherwise no file will be found*
  - **File names** *should not be hardcoded – they depend on the run number, on the source… → different at every run*

The TRIGGER Preprocessor

- *Following the request of the triggering detectors, a new preprocessor (the **Trigger Preprocessor**) has been introduced*
  - *Same logic as for any other preprocessor*
  - *Called at every run, but depending on the run type (as usual)*
  - *Reading the trigger mask from the DAQ logbook*
  - *Calling detector-specific procedures – one per triggering detector – as soon as the specific detector is included in the trigger mask*
    - *A special TRIGGERING SOR signal is sent to the triggering detectors to communicate with the FXSs*
    - *So far, only SPD has a non-empty procedure – thanks to A. Mastroserio*
    - *Any triggering detector needing to store trigger information in the OCDB should implement his own procedure*
  - *The output is stored in the OCDB, in the folder TRIGGER*
- *$ALICE_ROOT/TRIGGER module now present in AliRoot*

The GRP

Changes in the GRP

- ***Some of the DPs retrieved for the GRP are processed to get the statistical quantities to be stored in the OCDB***

  → *Mean, Median, Truncated Mean (+/- 3σ), Standard Deviation, Standard Deviation wrt Median*

- ***So far, a pure mean was calculated from the values collected (same for the other quantities)***

  → ***values were not properly weighed considering the DCS archival procedures***

  + ***not relevant in normal stable conditions***
  – ***giving problems when something was changing***

- ***A new procedure to calculate the statistical quantities has been implemented***

The new Procedure

- ***Based on the outcome of the discussion.***
- ***Dependent on:***

  nentries = n. of entries in [SOR, EOR] (SOR = DAQ_time_start, EOR = DAQ_time_end)
- ***Values weighed by the time difference between their own time stamp, and the time stamp of the subsequent value***

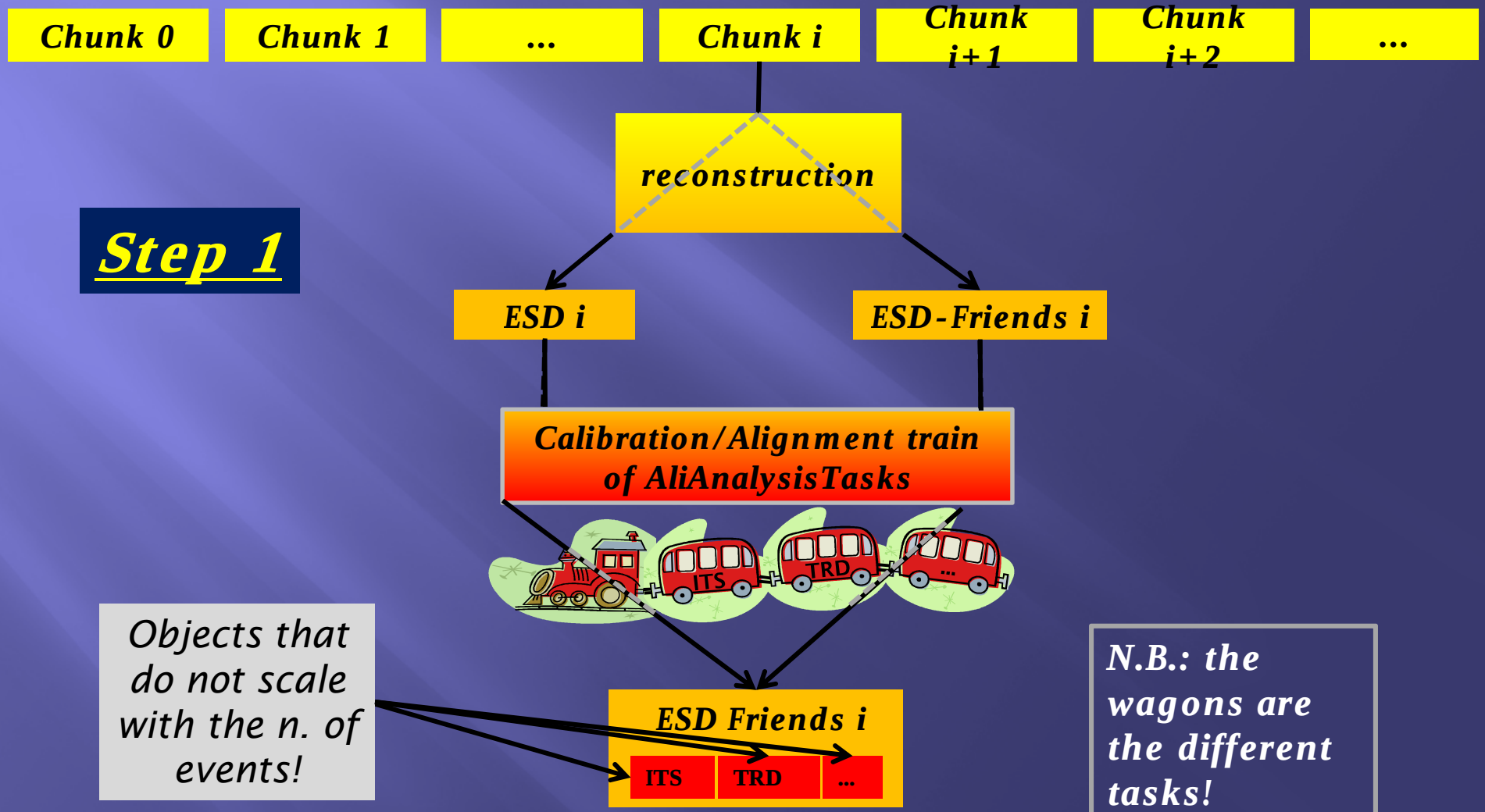
1. If (nentries > 1): all and only the values in [SOR, EOR] are used
2. If (nentries = 1): the only value in [SOR, EOR] is used, plus the last value recorded before SOR
3. If (nentries = 0): the last value before SOR is used

The GRP Failures at P2

- *Main causes for the GRP failure:*
  - *DAQ_time_start/end are not available →* **DAQ logbook error**
  - *Trigger scalers missing in DCS FXS →* **DCS FXS Error**
    - *Usually due to ecs_success flag missing in DAQ_logbook*
- *The current Shuttle logic makes these runs be processed*
- *The new Shuttle logic should avoid such errors*

The Offline Calibration/Alignment Framework

Calibration Step – 1



Features

- The reconstruction is left untouched
- Re-Use of the Analysis Framework with several tasks (Calib/Align)
- The tasks should have the same output file
- ESDFriends would be filtered → reduced in size
- The AliESDFriends file obtained from the reconstruction should be overwritten (→ on the Worker Node) with
  - Filtered friends
  - Objects in separte folders produced for calibration

Getting the Calibration Constants Step – 2

ESD 0	ESD ...	ESD i	ESD i+1	ESD ...
ESD Friends 0 (ITS, TRD, ...)	ESD Friends... (ITS, TRD, ...)	ESD Friends i (ITS, TRD, ...)	ESD Friends i+1 (ITS, TRD, ...)	ESD Friends... (ITS, TRD, ...)
AOD 0	AOD...	AOD i	AOD i+1	AOD ...

Merging/Analysis procedure to produce a final file

The expert...

automatically

N.B.: the wagons are the calibration objects written (in the ESD friends) before!

OCDB Updating

Step 3

The State of the Art

- *First implementation of the skeleton in progress*
  - **AliESDHandler**
    - *handler for the ESDs+ESDfriends output*
  - **AliAnalysisTaskFriend**
    - *deriving from AliAnalysisTask, very similar to AliAnalysisTaskSE*
    - *allows event selection*
    - *allows to add selected friends provided they're not already there*
  - **AliAnalysisTaskCopyESD**
    - *Test task to copy ESDs → output stored in AliESDs_v1.root file*
  - **AliAnalysisTaskFilterFriend/FriendSecond**
    - *Test tasks to filter friends both at the level of the track and of the event*

Features

- ***The filtering procedures require to write null pointers in the ESD Tree – at the level of ESDfriend object, and of ESDfriendTrack in the TClonesArray of the ESDfriend***
- ***This is possible only setting the split mode of the ESDfriend branch to zero*** → *The gain of the filtering procedure must compensate the lost in compression with splitmode=0*
- ***To write null pointers in the TClonesArray of AliESDfriendTracks, the BypassStreamer for the TClonesArray has to be set to kFALSE***
- ***A new function*** AliESDfriend::AddTrackAt(int_t index) ***has to be introduced***

Adding Objects to AliESDfriends_v1.root

- *Apart from filtering, the possibility to **add objects** that don't scale with the number of events (histograms) is foreseen*
  - *These objects should be stored in the new AliESDfriends.root file in separate folders*
    - *Functionality to create folders in output files now there (see A. Gheata's talk from yesterday)*
- **AliAnalysisTaskAddObject**
  - *Deriving from AliAnalysisTask, adds an histogram to the AliESDfriends_v1.root file → still under test*

Results

- *First tests (local) give successful results*
- *Further improvement and tests required*
  - *Integration of existing tasks*
- *Many interactions needed with Offline/ROOT people*

  → Many thanks to Rene, Cvetan, Andrei, Peter, Andreas for their precious help!

- *Still a long way to go.....*

Backups

Calibration Step – 1 – Some Details

For chunk i....