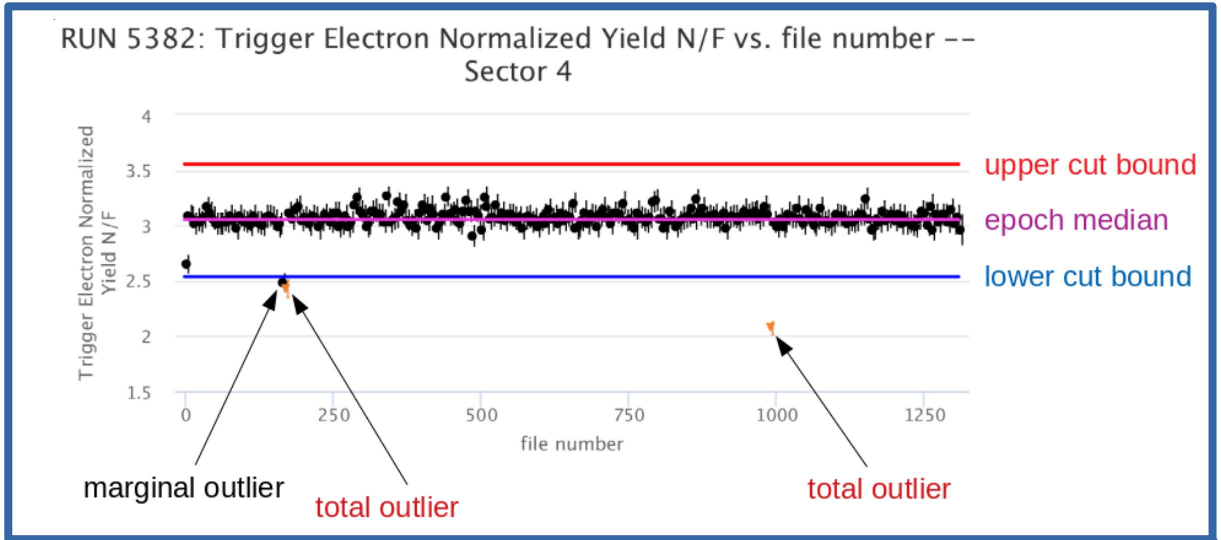
CLAS12 Quality Assurance

Christopher Dilks
CLAS Collaboration Meeting
4 June 2021

Quality Assurance Timelines

Monitor helicity sign

Identify outliers of # trigger electrons, normalized to FC charge



upper cut bound

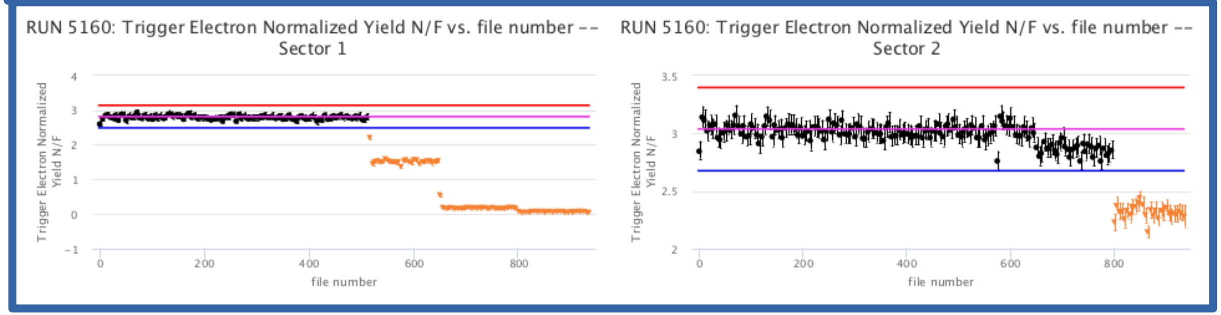
epoch median

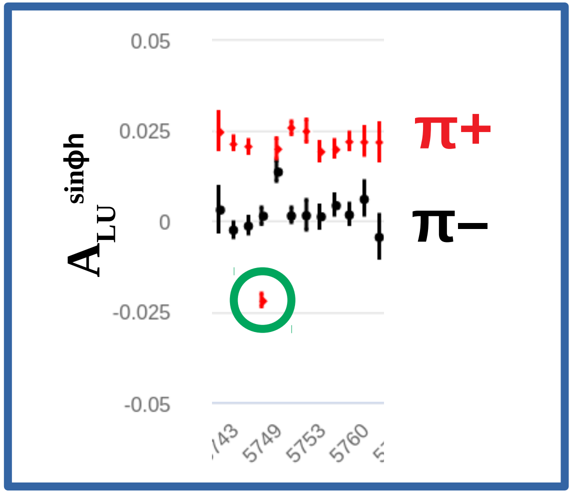
lower cut bound

marginal outlier

total outlier

total outlier



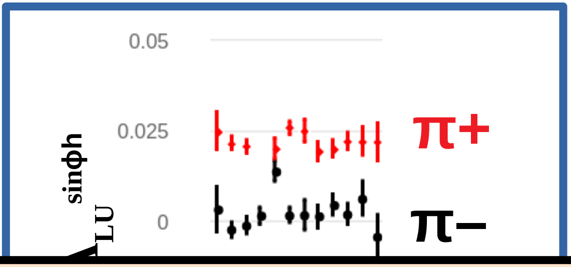


Document other miscellaneous cases:

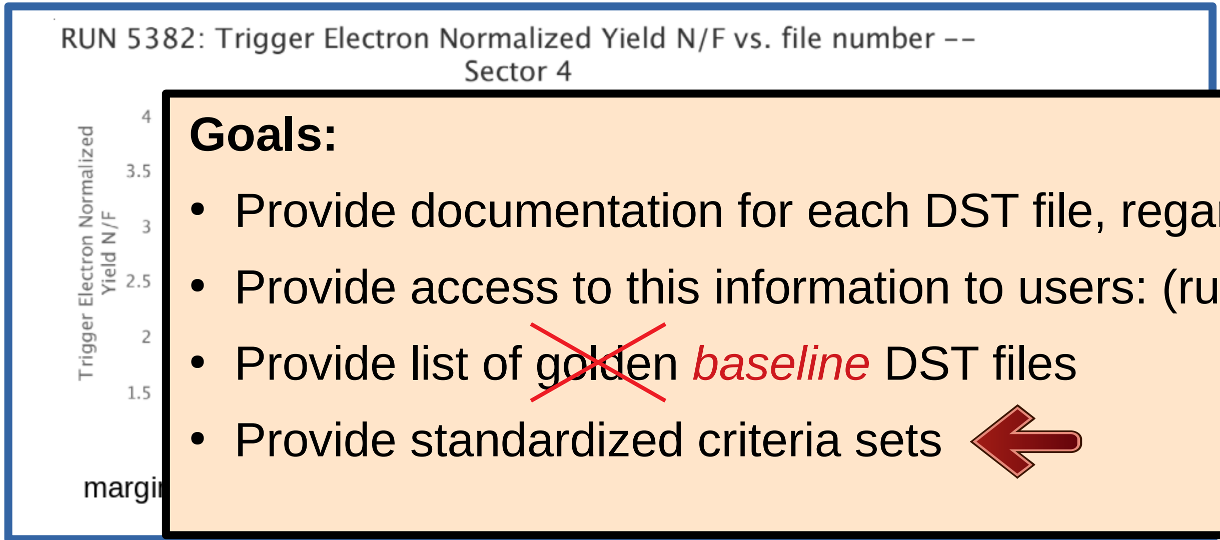
- special runs, e.g., low lumi
- junk runs
- anything else noteworthy

Quality Assurance Timelines

Monitor helicity sign

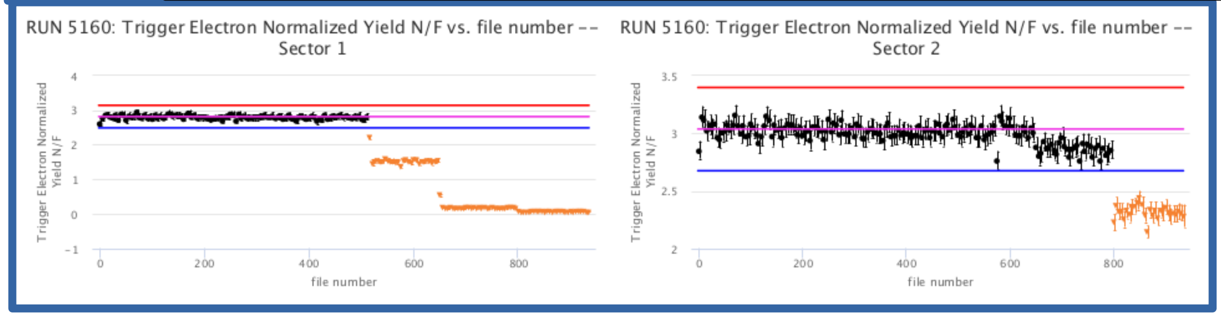


Identify outliers of # trigger electrons, normalized to FC charge



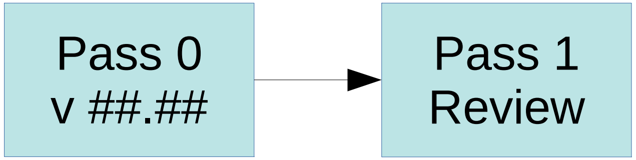
Goals:

- Provide documentation for each DST file, regarding QA status
- Provide access to this information to users: (runNum, eventNum) → QA status
- Provide list of ~~golden~~ *baseline* DST files
- Provide standardized criteria sets

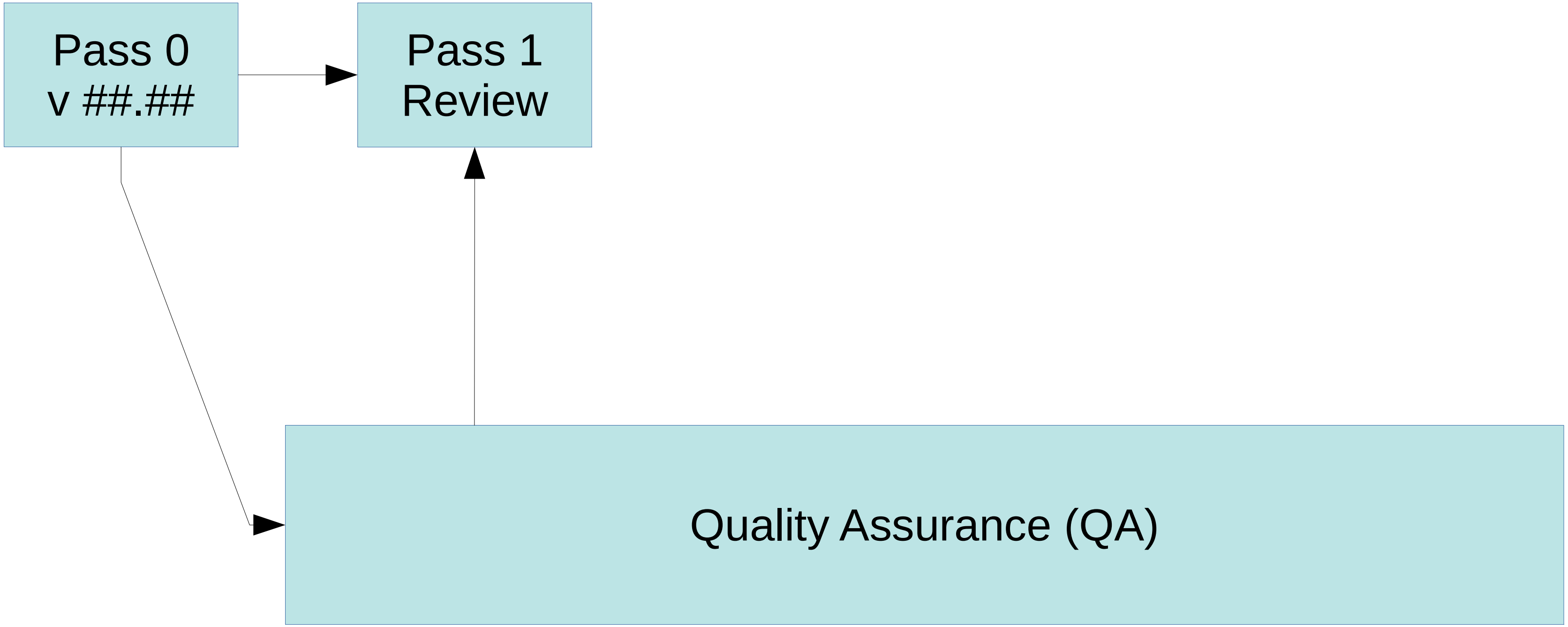


- special runs, e.g., low lumi
- junk runs
- anything else noteworthy

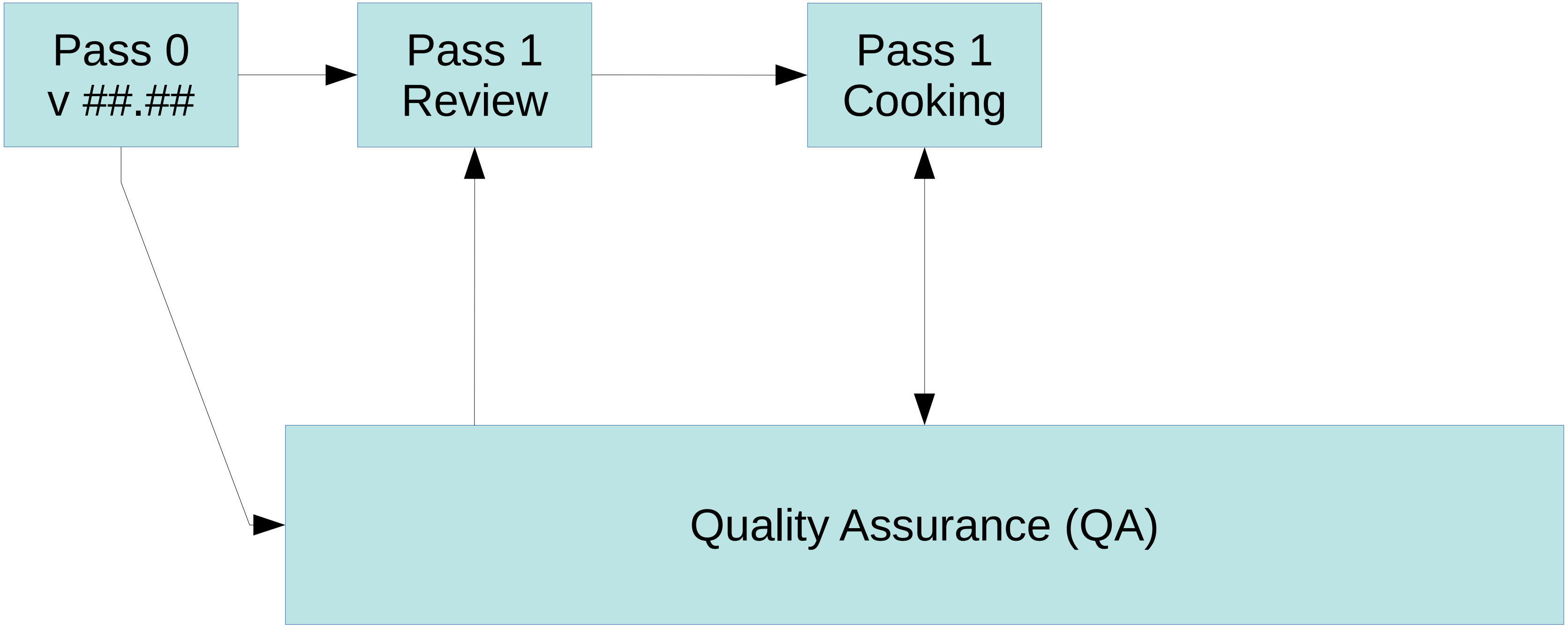
The Procedure



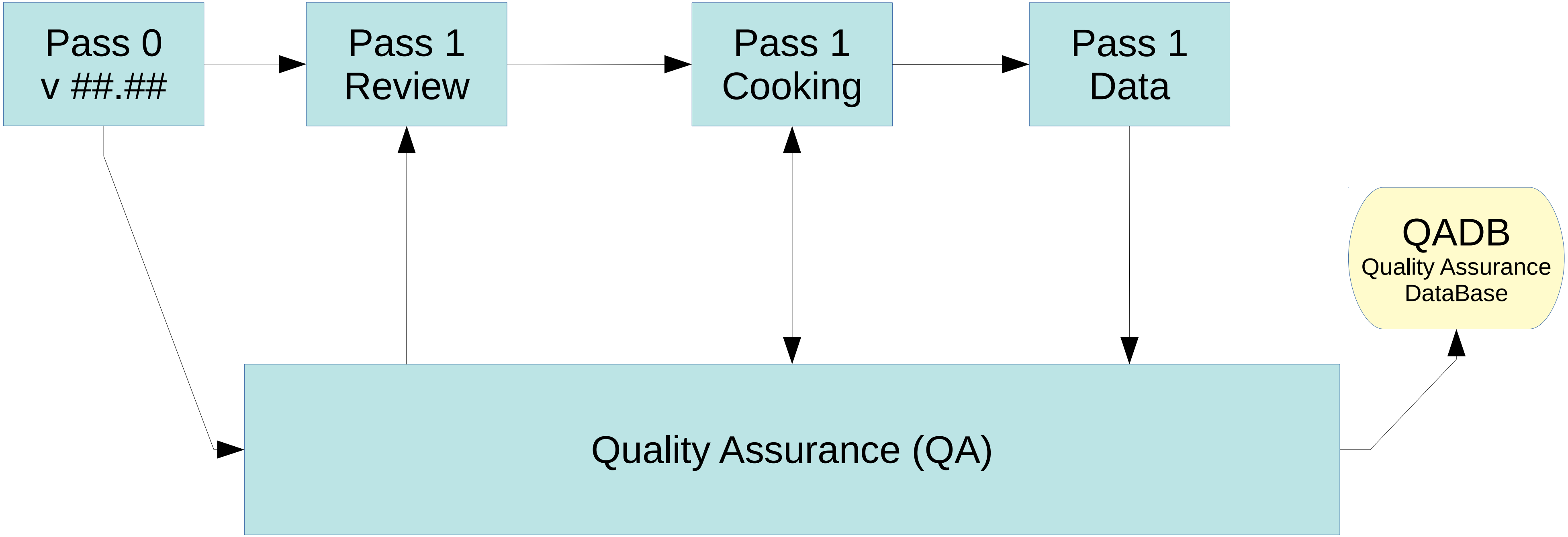
The Procedure



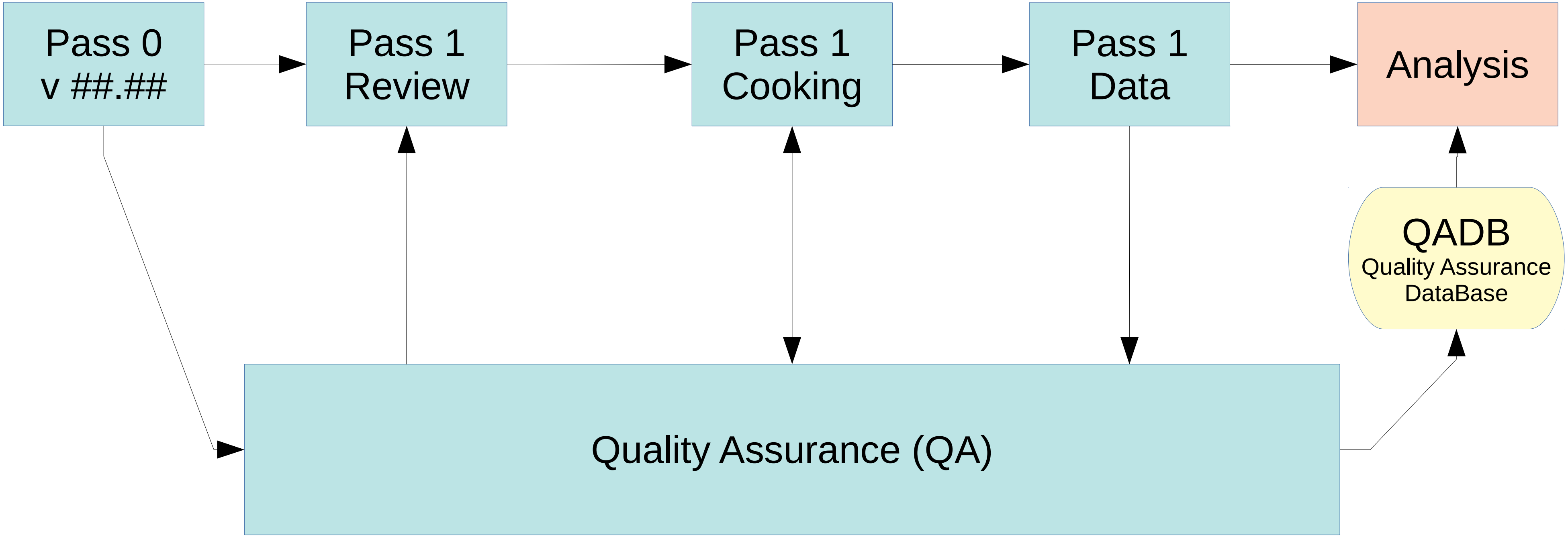
The Procedure



The Procedure



The Procedure



People Power

Development

Chris Dilks

Lucilla Lanza

Matthew McEneaney

Support

Andrey Kim

Nick Markov

Latifa Elouadrhiri

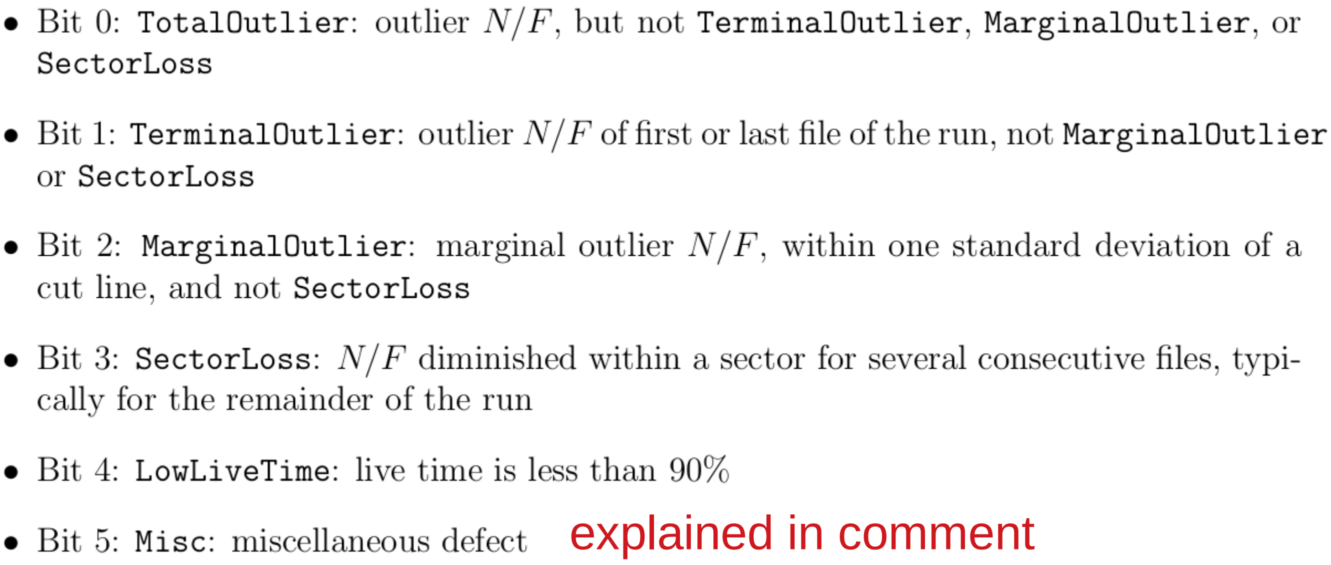
Collaboration feedback and advice

Documenting QA Status: the QADB

Each DST file can be assigned ***defect bits***, along with an optional comment

Defect Bits:

FD electron outliers



- Bit 0: `TotalOutlier`: outlier N/F, but not `TerminalOutlier`, `MarginalOutlier`, or `SectorLoss`
- Bit 1: `TerminalOutlier`: outlier N/F of first or last file of the run, not `MarginalOutlier` or `SectorLoss`
- Bit 2: `MarginalOutlier`: marginal outlier N/F, within one standard deviation of a cut line, and not `SectorLoss`
- Bit 3: `SectorLoss`: N/F diminished within a sector for several consecutive files, typically for the remainder of the run
- Bit 4: `LowLiveTime`: live time is less than 90%
- Bit 5: `Misc`: miscellaneous defect

explained in comment

Under construction:

- FT defect bits → Lucilla Lanza
- Pion outlier bits / generalization → Matthew McEneaney
- CD outlier bits

QADB for the Users

https://github.com/JeffersonLab/clasqaDB

Included Data Sets

- RG-A
- RG-B
- RG-K
- started RG-F

- Online Timelines → visual access
- Human-readable Tables → quick lookup for browsing
- Software access for analysis
  - C++ (directly, or via *clas12root*)
  - Groovy
  - To do: Fortran (help!)

QADB also provides:
Faraday Cup charge, filtered by the QA
cross-checked between Chris & Lucilla

To use the software, user only needs **run number** and **event number**; QADB software will find the associated DST file and return corresponding QA information

QADB for the Users

Principle of Least Action

$$\frac{\delta S}{\delta \mathbf{q}(t)} = 0$$

QADB Information Provided

Outlier FD electron

Low Live Time

Misc

Comments

Simplified Booleans

OkForAsymmetry

Used for RGA 1st publications of beam spin asyms

Golden

“Golden” = NO defects, perfect DST files

N.B.: Most “good” runs are ~98% golden, *very few* are 100% golden

QADB for the Users

Principle of Least Action

$$\frac{\delta S}{\delta \mathbf{q}(t)} = 0$$

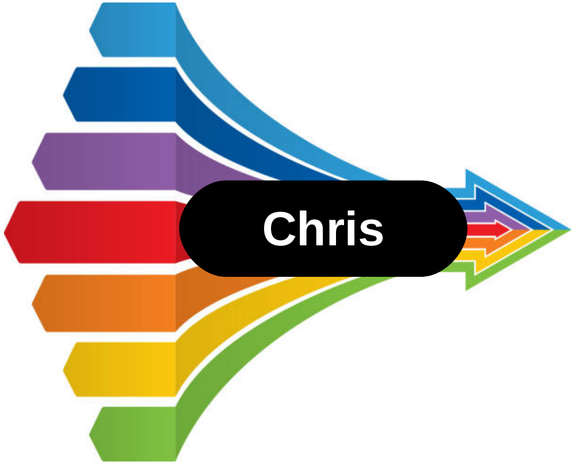
QADB Information Provided

Outlier FD electron

Low Live Time

Misc

Comments



Simplified Booleans

OkForAsymmetry

Used for RGA 1st publications of beam spin asyms

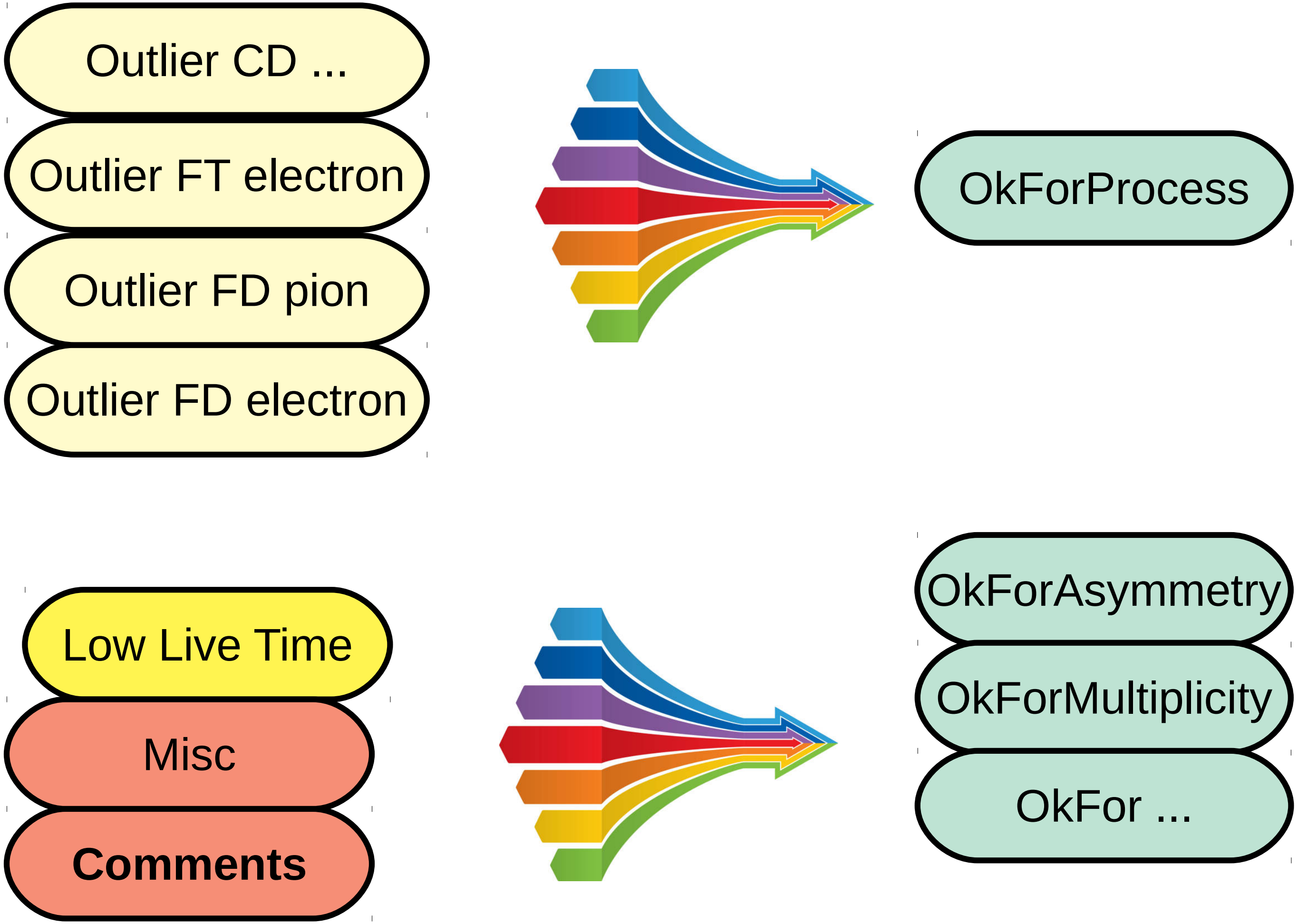
Golden

“Golden” = NO defects, perfect DST files

N.B.: Most “good” runs are ~98% golden, *very few* are 100% golden

QADB Upgrades

Simplified Booleans



Outlier CD ...

Outlier FT electron

Outlier FD pion

Outlier FD electron

OkForProcess

Low Live Time

Misc

Comments

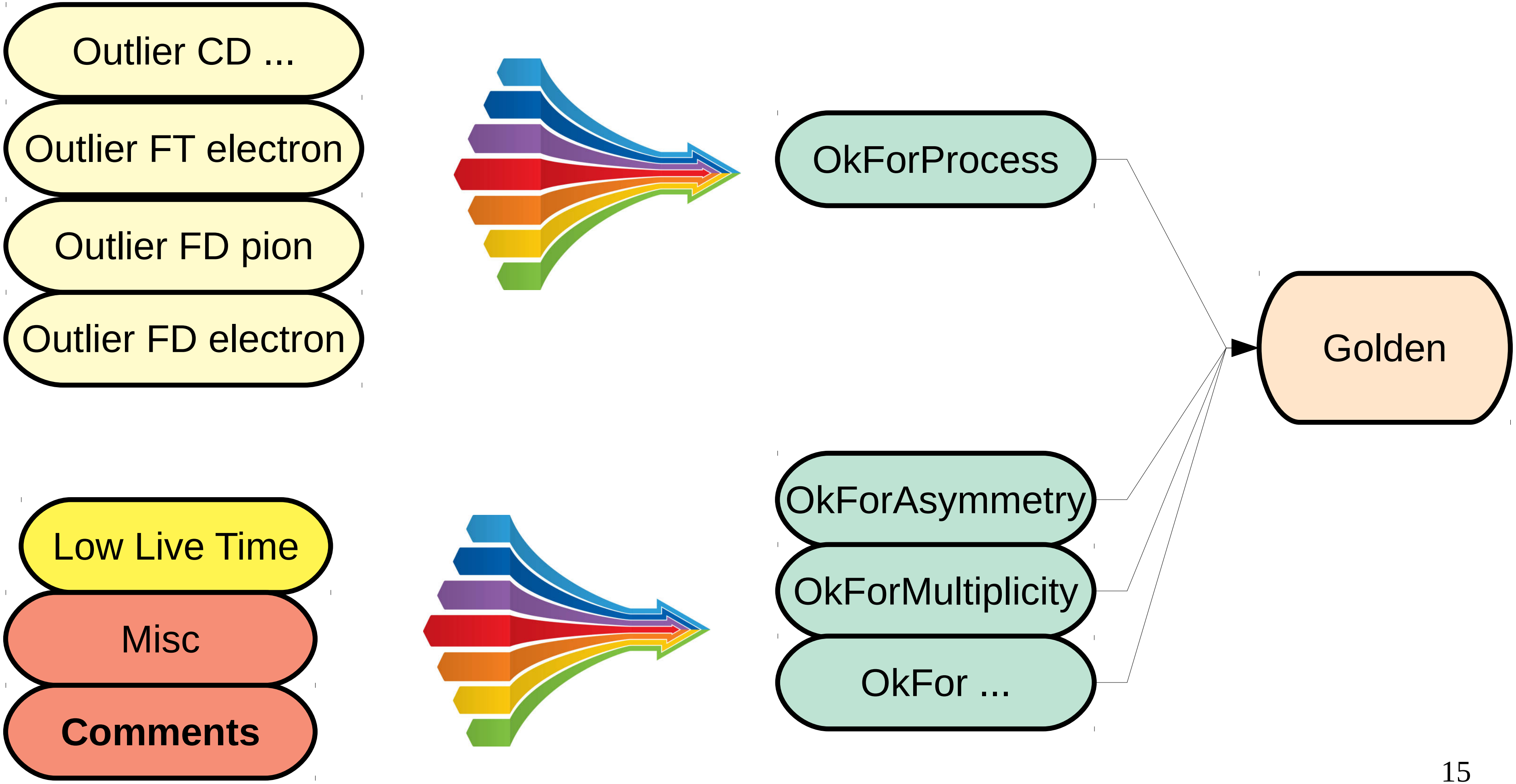
OkForAsymmetry

OkForMultiplicity

OkFor ...

QADB Upgrades

Simplified Booleans



Outlier CD ...

Outlier FT electron

Outlier FD pion

Outlier FD electron

Low Live Time

Misc

Comments

OkForProcess

OkForAsymmetry

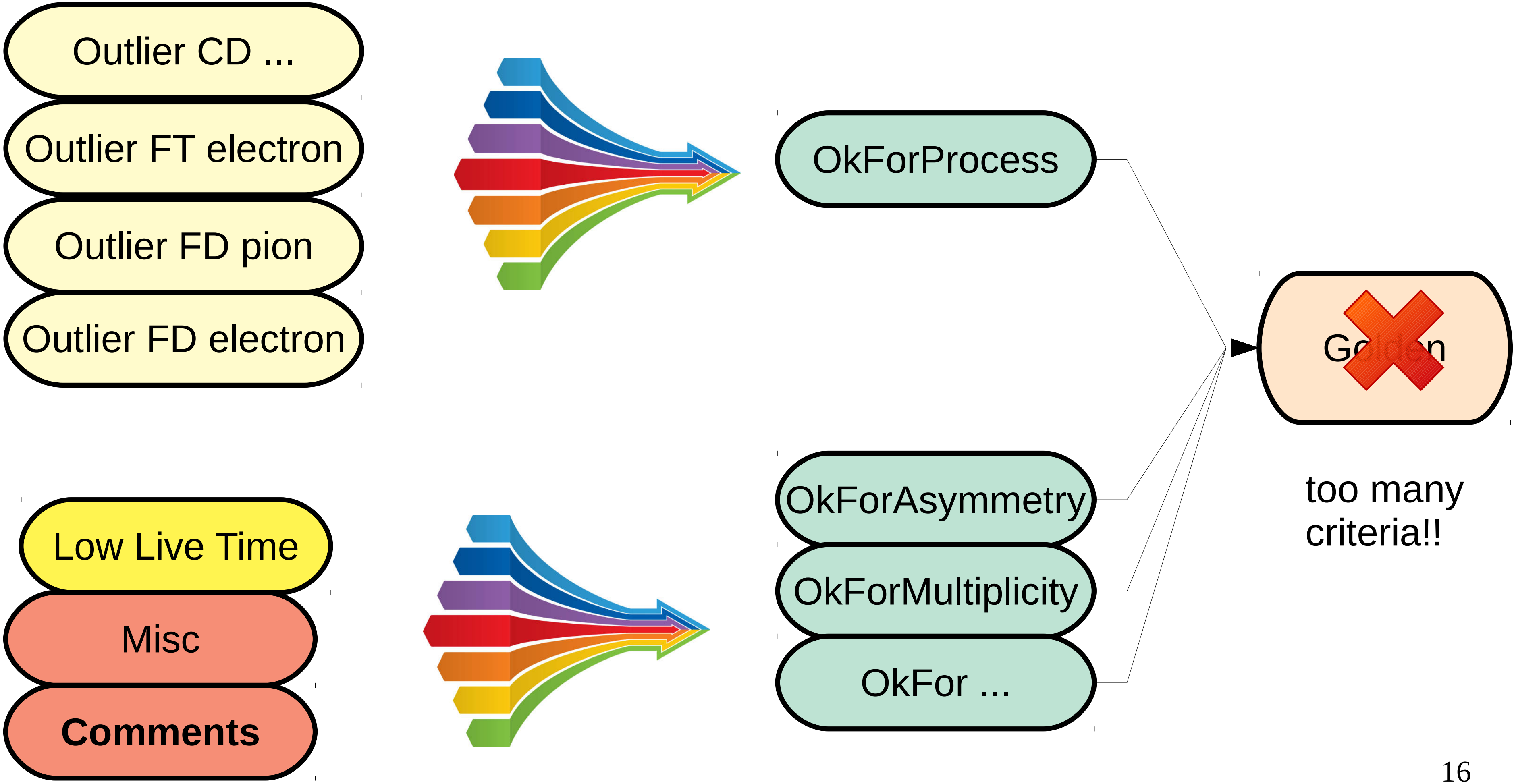
OkForMultiplicity

OkFor ...

Golden

QADB Upgrades

Simplified Booleans



Outlier CD ...

Outlier FT electron

Outlier FD pion

Outlier FD electron

Low Live Time

Misc

Comments

OkForProcess

OkForAsymmetry

OkForMultiplicity

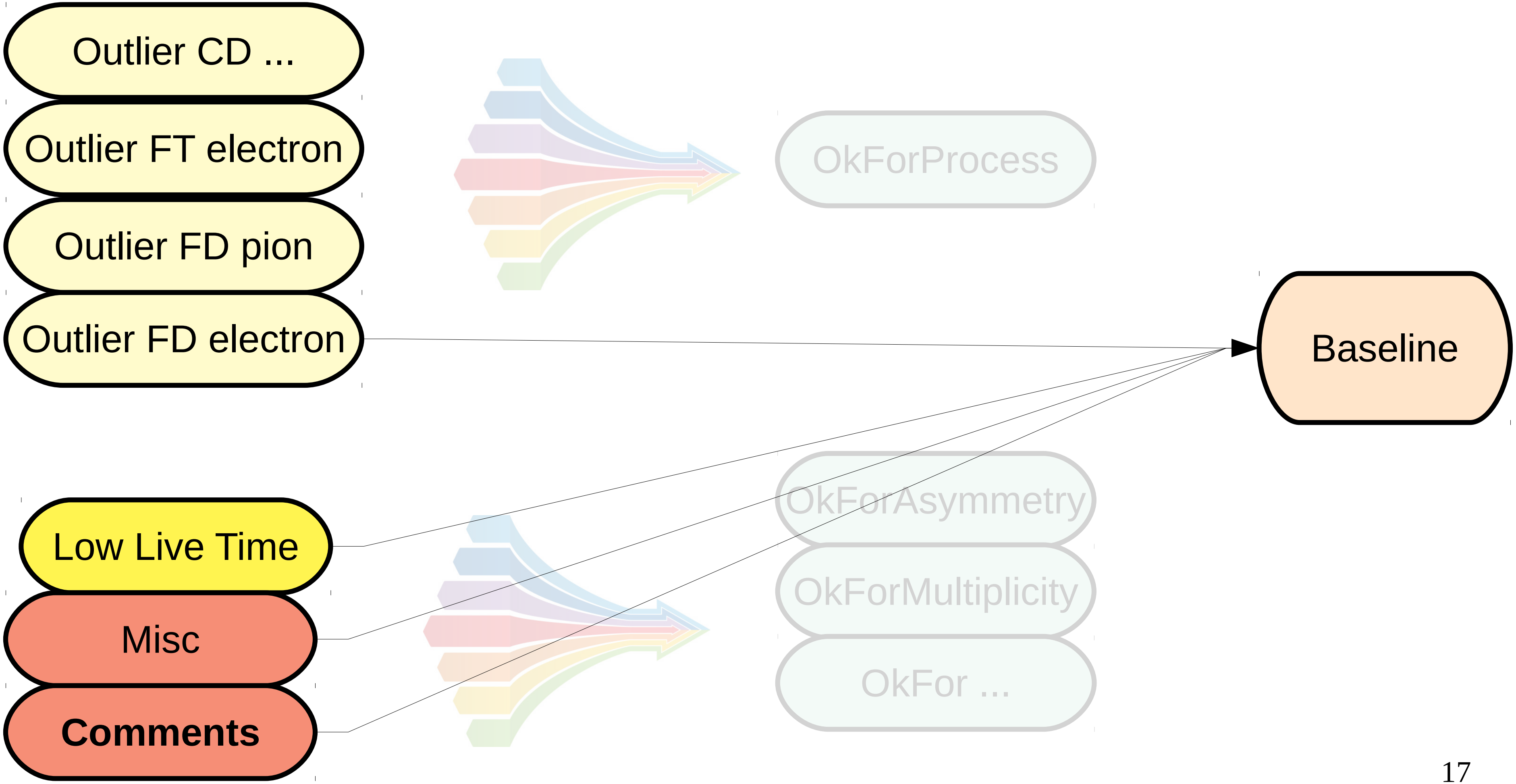
OkFor ...

~~Golden~~

too many
criteria!!

QADB Upgrades

Simplified Booleans



Outlier CD ...

Outlier FT electron

Outlier FD pion

Outlier FD electron

Low Live Time

Misc

Comments

OkForProcess

Baseline

OkForAsymmetry

OkForMultiplicity

OkFor ...

QADB Upgrades

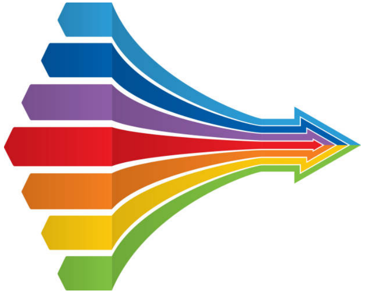
Simplified Booleans

Outlier CD ...

Outlier FT electron

Outlier FD pion

Outlier FD electron



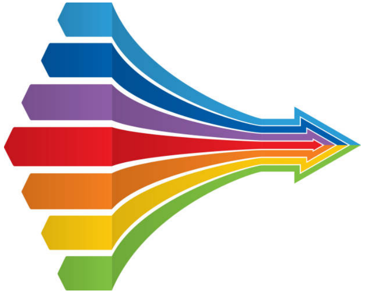
OkForProcess

Decision-Making Responsibility

Low Live Time

Misc

Comments



OkForAsymmetry

OkForMultiplicity

OkFor ...

note: not all OkFor_____ files will be “baseline”

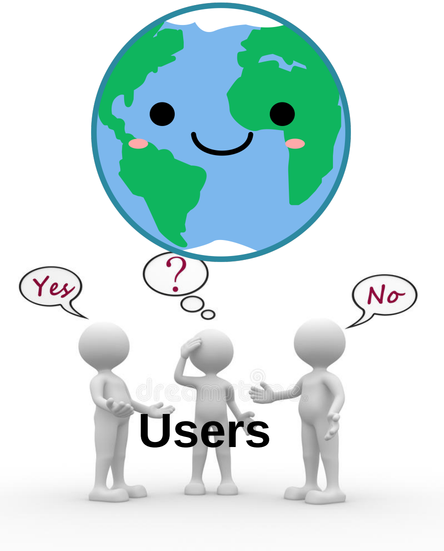
Example Comments in the QADB, for Misc Defect Bit

- all ROC rates at zero; ended run
- beam current ramping
- **beam current was drifting; sporadic livetime issues**
- DAQ test, omit this run
- DC trips
- **empty target**
- FC charge offset problem
- **fluctuation in pion pT and theta**
- fraction of events with defined helicity is low
- hardly any beam
- **helicity sign is wrong**
- junk run
- livetime issues; logbook mentions zig-zag problems in the current
- **low lumi run**
- N/F abnormally high
- N/F is high for the whole run
- no connection to ECAL sector 2, but N/F looks normal
- no helicity info
- RGK trigger test
- ROC problems near the end of this run
- sector 1 was lower than normal, but not by much; marked as sectorLoss anyway
- short and trippy run
- **used different trigger file, rgb_v9_1.cnf; N/F lower than usual; Beam current changed to 50nA**
- zero-field alignment run
- …

Who judges whether “Misc” = bad or good?

Bearing the QA Burden

More Data
More Users
More Analyses



Users

Experts

Decision-making Responsibilities

- Determine analysis **categories**, which group analyses together with common goals
- For each category/observable, define a **QA criteria set**, which can be applied to each analysis of the category
- Facilitate **communication** between QA developers and the needs of the collaboration
- Long term responsibility, but not a time-consuming effort
- Need input from **diverse** group of experts
- QA Procedure and usage of QADB is **optional** for any Run Group and/or analysis, but recommended

The Way Forward…

- Committee?
- Task Force?
- Subgroup within existing Committee or Task Force?
- Expand responsibility of QA developers, bringing in expert help for decision-making?
- Do not provide “simple” booleans, forcing users to make decisions?
- Do nothing, leaving all as is?

bad ideas...

backup

Example Criteria Set: OkForAsymmetry

Current QA Implementation

Defect Bits:

FD electron outliers

Reject
- Bit 0: `TotalOutlier`: outlier N/F, but not `TerminalOutlier`, `MarginalOutlier`, or `SectorLoss`

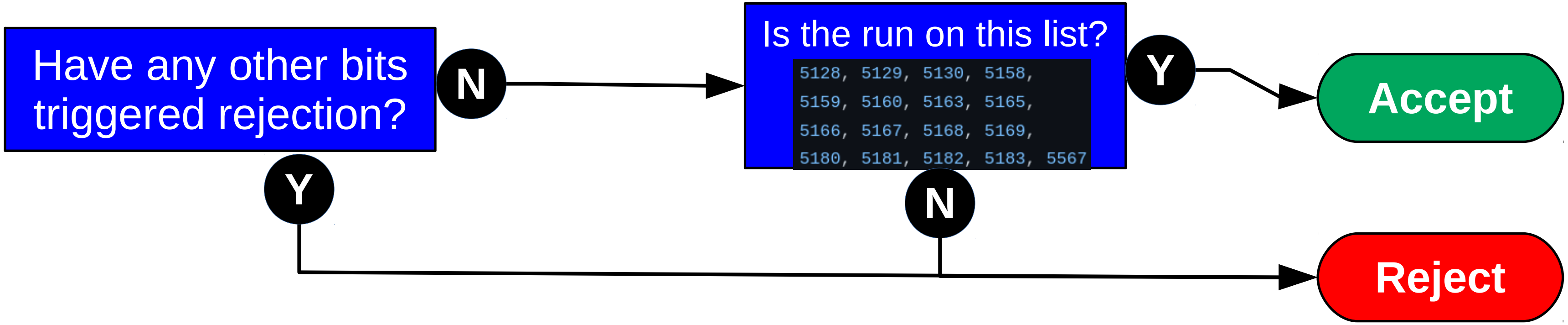
Reject
- Bit 1: `TerminalOutlier`: outlier N/F of first or last file of the run, not `MarginalOutlier` or `SectorLoss`

Reject
- Bit 2: `MarginalOutlier`: marginal outlier N/F, within one standard deviation of a cut line, and not `SectorLoss`

Reject
- Bit 3: `SectorLoss`: N/F diminished within a sector for several consecutive files, typically for the remainder of the run

Accept
- Bit 4: `LowLiveTime`: live time is less than 90%

Conditional
- Bit 5: `Misc`: miscellaneous defect



```
5128, 5129, 5130, 5158,
5159, 5160, 5163, 5165,
5166, 5167, 5168, 5169,
5180, 5181, 5182, 5183, 5567
```

QA Decision-Making Plan

- Define list of analysis categories (observables)
  - Need input from:
    - Run group chairs
    - Physics working group chairs
    - Process chairs (e.g. Harut for SIDIS)
- Decide whom to invite to the group
- Kick-off meeting
  - Introduce goals, what we are doing, and why
- Develop spreadsheet of defect bits & misc cases vs. analysis category
  - This spreadsheet will be the basis of the subsequent QA Criteria Review Meeting
  - Group together common misc. cases
- QA Criteria Review meeting
  - Each run group category will individually decide:
    - Yea/Nay for each defect bit
    - Yea/Nay for each misc. case run
- Implement in QADB, and release to the collaboration!

Long Term Support Commitment

- Future data and analysis can bring:
  - New defect bits
  - New misc. cases
  - New passes (e.g., pass2) could also raise new QA concerns

- Responsibility for decision-making should new questions arise
  - After the QA Criteria Review meeting, QA developers will have a much clearer idea of the expectations of quality
  - Utilize past QA requirements to make decisions on future data
  - … but there could always be new unforeseen cases which will require input from experts (likely solvable over an email / shorter meetings)