



# Data Processing Update

Data Production meeting  
19/03/2020

Stefano Lacaprara, Marco Milesi

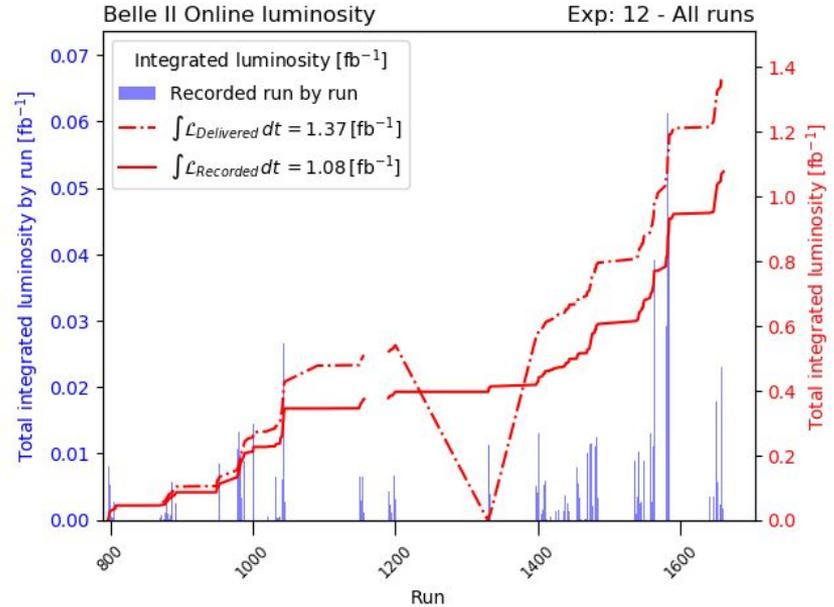
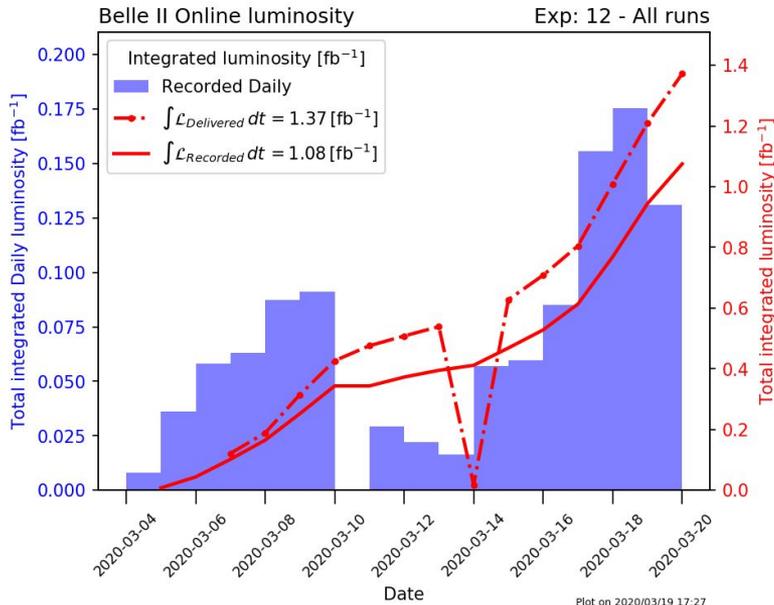
INFN Padova, Uni Melbourne



# Exp 12 status



- Reached 1.08 /fb recorded lumi (1.37 /fb delivered)
- Lumi plot using RunDB (with some issues)
  - Produced automatically every 12h
    - <https://confluence.desy.de/display/BI/Phase+3+luminosity+monitoring>
    - <https://confluence.desy.de/display/BI/Belle+II+Luminosity>



# Exp12 Unofficial



- As usual, started unofficial processing using **GT = online**
  - <https://agira.desy.de/browse/BIIDP-2461>
  - Only for Skims = ['mumu\_2trk', 'bhabha', 'gamma\_gamma', 'hadron', 'bhabhaec1', 'radee']
  - Cdst output
- Raw skims produced by CC (new from exp12!)
  - <https://agira.desy.de/browse/B2CC-205>
  - Needed some minor adjustment to DP script <https://stash.desy.de/projects/B2P/repos/data/pull-requests/106/overview>
- Runs COMPLETED, N = 138/173 ; 35 Running.
  - Path: /group/belle2/dataprod/Data/Unofficial/e0012/

# Upgrade in DP scripts



- Better handling of log files (avoid too many log in one directory)
- Better integration with Airflow (w/ David)
  - Have Airflow use the very same script used by DP, to avoid code duplication and ensure coherence
    - Check out in Airflow well-defined Git tags of the DP tools right before processing, and import steering scripts from there → coordinate on appropriate naming scheme.
    - David is testing the workflow in a dummy prompt processing now...
- Several minor fixes, improvements (mostly for DP managers)
- TODO (next slides)
  - Easier handling of special, dedicated production
    - Several have been requested in this period
  - Review the DataObj the we are currently adding to m/cdst in DP script
  - Work for a solution against zero-events input files

# Handling of special processing



- Especially those requiring modification of production scripts
  - Want to have possibility to add via json (DP configuration file) DataObj to DataStore
    - Now need to change by hand the python script <https://agira.desy.de/browse/BIIDP-2496>
  - GT set definition already configurable easily, but not enough.
  
- Hard to cope with all possible scenarios
  - Eg: recent requests for:
    - No-hlt trigger filtering
    - Process cosmic/beam as physics
    - RAW dataobjects to be added to the output
  
- How many special processing will be requested?
  - Too many ad-hoc processings could easily create bookkeeping havoc...

# DataObj in DataStore



- Ideally, we'd like to have no additional DataObj added to the output file by DP steering scripts, but rather have all the needed one defined by default by release helper functions. <https://agira.desy.de/browse/BIIDP-2495>
  - This is not the case today.
  - <https://stash.desy.de/projects/B2P/repos/data/browse/Proc/rec/runRecPhysics.py>
- Input
  - `raw_branches = ["EventMetaData", "ROIs", "ROIpayload", "RawPXD", "RawSVDs", "RawCDCs", "RawECLs", "RawARICHs", "RawKLMs", "RawTOPs", "RawTRGs", "SoftwareTriggerResult"]`
  - `main_path.add_module("RootInput", inputFileNames=inFileNames, branchNames=raw_branches)`
- Mdst output
  - `add_mdst_branches = ["Muids", "TracksToMuids"]`
- Cdst output
  - `add_cdst_branches = ["SVDS Shaper Digits From Tracks", "TRGGDL Unpacker Stores", "VXDDedx Tracks", "VXDDedx Likelihoods"]`

# Additional module parameter



- Several module parameter are set in DP script:
  - Need to review to be sure they are still needed

```
# Patch by Sagar Hazra - BII-5051
```

```
basf2.set_module_parameters(main_path, "VXDDedxPID", enableDebugOutput=True)
```

```
# Patch by Maiko Takahashi - BIIDP-1366
```

```
basf2.set_module_parameters(main_path, "PXDPstErrorChecker", CriticalErrorMask=0)
```

```
# Silence PXDUnpacker warnings - BIIDP-1531
```

```
basf2.set_module_parameters(main_path, "PXDPstErrorChecker", logLevel=basf2.LogLevel.ERROR,  
SuppressErrorMask=0xFFFFFFFFFFFFFFFF)
```

```
# Patch by Koga-san + Ritter-san:
```

```
# exclude the TRGCDCT3DUnpackerModule from the path.
```

```
final_path = tools.remove_module(main_path, module="TRGCDCT3DUnpacker")
```

# Zero event output/input files



- Currently we can have 0-events output files.
  - If we read them as input, we get a **B2FATAL**
  - **This is problematic for DP:**
    - Not really a failure to be investigated, but still appears as such
    - Not so rare for short runs and hard skimming (eg hlt\_skim or analysis skim)
    - Now CC is copying RAW hlt skims on the grid:
      - If they are empty, a lot of trouble will follow.
  - What do we want to do with these files?

# How to cope with zero-events files



- Options (my personal view)
  - We can do at basf2 level or at processing level
- **basf2**
  - **Avoid B2FATAL altogether**
    - Exit, but with a more gentle error.
    - Processing is fine, no output is produced.
      - There might be reason to avoid this in basf2, don't know
  - **Do not write output if 0-events will be written.**
    - Might be problematic, since I guess basf2 create the file at the beginning, and then fill it, it would need to delete the file if not filled, bleah!
    - Also, not sure if grid processing is fine to have a job w/o any output.

# Zero-events handling (cont'ed)



- **Processing:**

- **Easy solution:** before load a file (or more) as input, check if contains no events. If so, skip processing.
  - Possibly reading file metadata (?)
  - Our poor-man solution → simply add all input(s) in a TChain, and GetEntries() (quick!)
    - (SL: this is a nasty workaround, that we should try to avoid if at all possible)
  - Same problem as before: we might have jobs w/o output.
    - **Grid processing: is it fine to have a jobs which produces no output at all?**
- **Not-so easy:** After any processing (eg hlt/analysis skim) check if file is empty, if so, do not pass to next step.
  - Requires an additional step at end of basf2 processing
    - Easy locally, possibly not so on the grid.
    - Possibly just reading the output file metadata (?)
  - What to do if the output file is empty?
    - Do not copy to final location (local processing)
  - What about grid processing? Skip file for merge step? Copy to SE and invalidate in AMGA?
    - Small, useless file to pollute SE and catalogue, really?

# Bucket9 and proc11



- What are the plan for those?
- Bucket9
  - We have today 1/fb, might reach 2/fb soon
- Proc11
  - Using releare 04-02-00 ?
  - And new cdc calibration
  - timescale?