EventIndex

Status and Developments

Dario Barberis

Genoa University/INFN

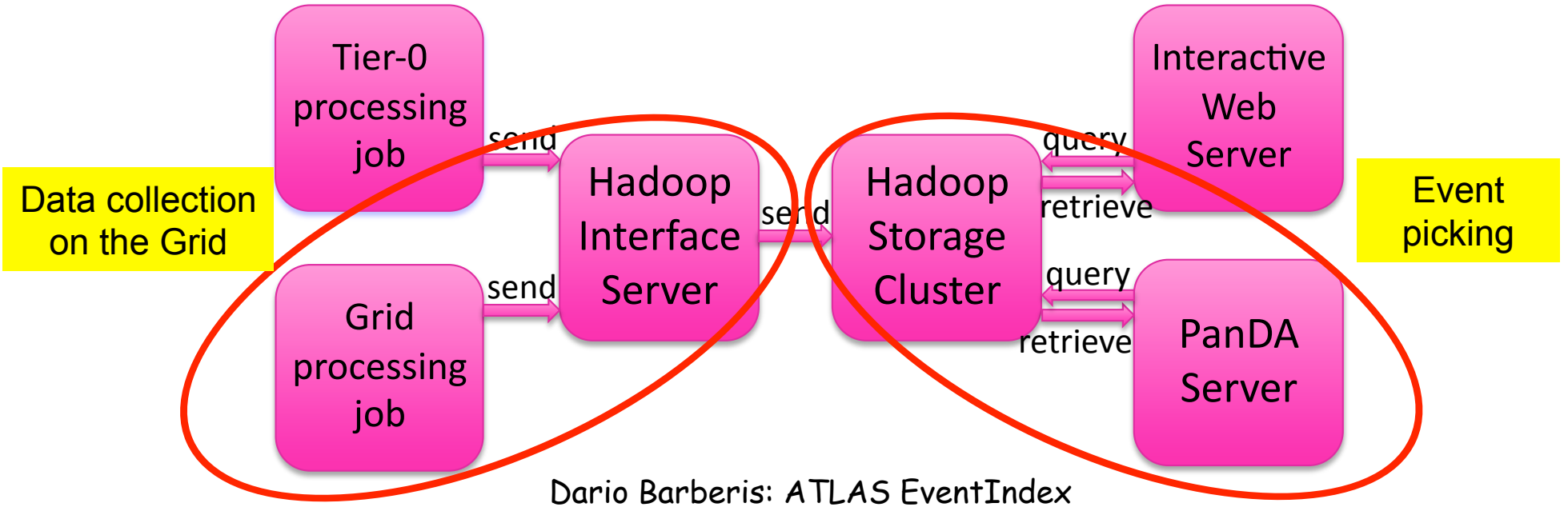
On behalf of the EventIndex group

Topics

- Project status and developments
  - Data collection (Tier-0 and Grid)
  - Data validation
  - Hadoop storage
  - Trigger decoding
  - Monitoring
- "Last mile"
  - Event picking
  - Duplicate event finding
  - DAOD overlaps

EventIndex Project Breakdown

- We defined 4 major work areas (or tasks):
  1) Core architecture
  2) Data collection and storage
  3) Query services
  4) Functional testing and operation; system monitoring

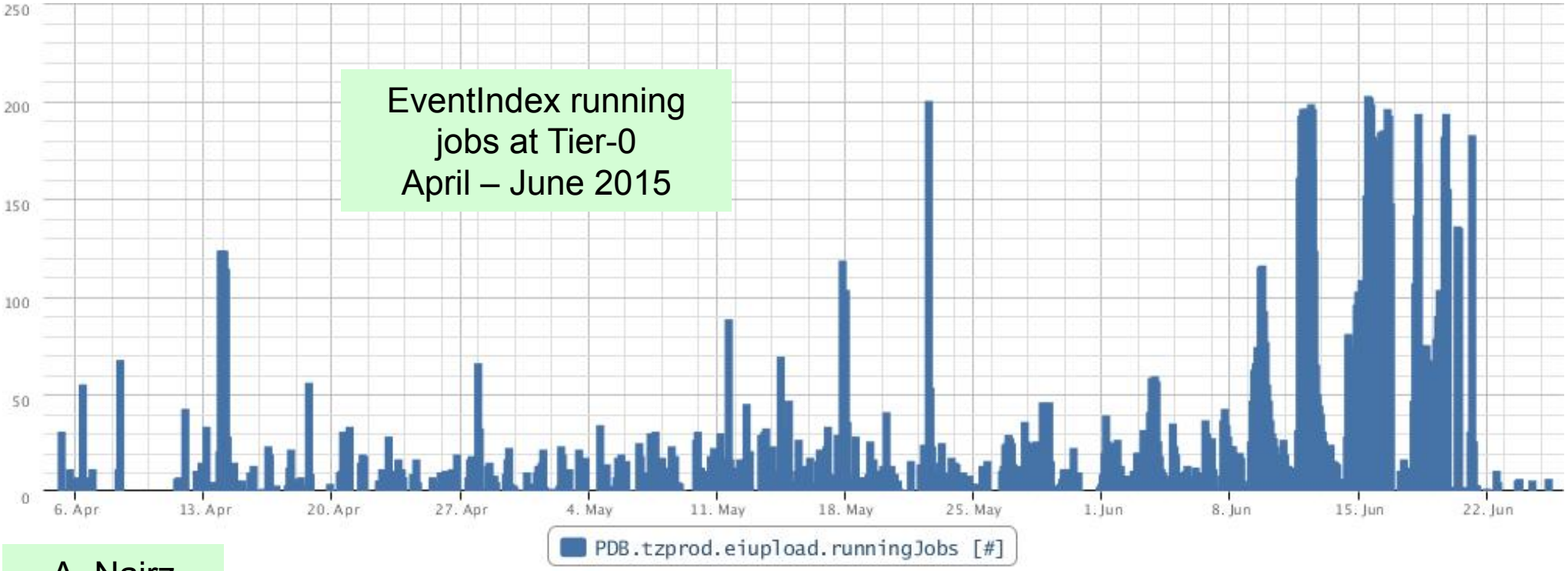


EventIndex Workshop 22-23/06/2015

- Two full days and evenings of discussions and hands-on work in the nice climate and pleasant environment of Valencia
  - Remaining developments for the current data-taking operations
  - Operation issues and possible solutions
  - Longer-term plans
- Thanks very much to our hosts at IFIC-Valencia!

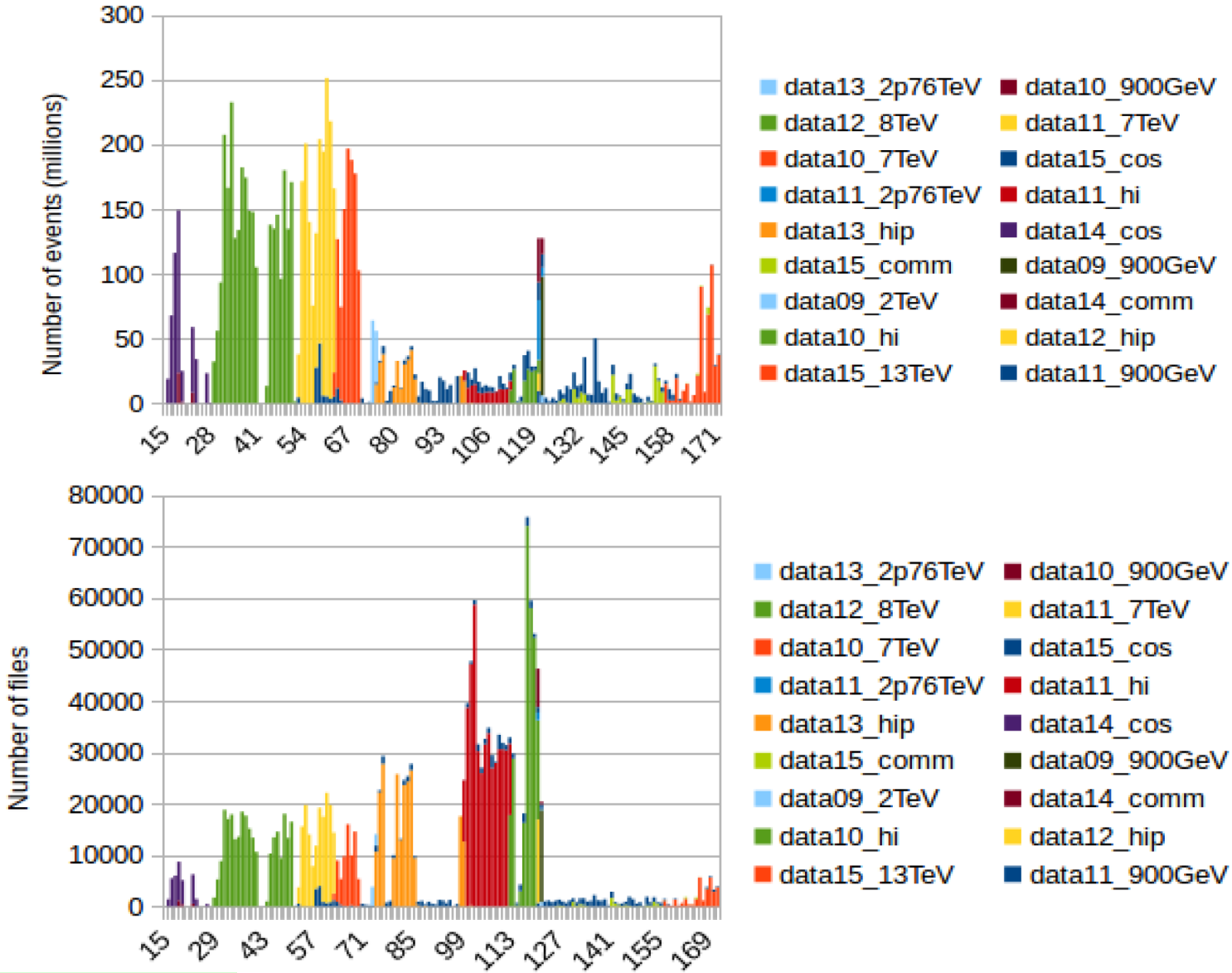
Data Collection: Tier-0 jobs

- Tier-0 runs since several months smoothly EI Producer jobs on the merged AOD and derived datasets
  - ✓ No real problem observed so far
  - ■ But derived data indexing not yet started
- Internal validation of received data processed as fast as currently possible
  - ■ see later



A. Nairz

Data Collection: Tier-0 jobs



J. Sanchez, A. Fernandez

Data Collection: Grid jobs

- Grid (re)processing of real and simulated data: get EI info from "permanent" EVNT, AOD and DAOD for all data.
  - EVNT to feed the Event Service
  - AOD and DAOD for all other use cases
  - They are all POOL/ROOT files so the same transformation can be run on all datasets
- Agreements during/after discussions last months:
  1) Base the dataset selection on AMI knowledge of dataset status
     - This is the physicists view of available datasets for analysis so we choose to keep in sync with it
       - Datasets may appear and disappear from the list of "good" datasets
       - But see next slide
  2) Keep dataset metadata in the EI store for consistency checks and status updates
  3) Use the "open-ended production" mechanism to automatically generate EI tasks when new datasets are available

Dataset selection

- Cron task runs nightly and uses AMI client to
  - Get the list of VALID datasets with:
    - projectName = mc15% or data15%
    - dataType = EVNT or %AOD%
    - prodsysStatus = "ALL EVENTS AVAILABLE"
    - lastModificationTime = (yesterday)
  - Retrieve for each dataset:
    - Dataset name
    - Number of files
    - Number of events
    - List of files
  - Store this info in EventIndex space
    - HBase looks like the simplest solution for the time being
  - ✓ Working
    - Most of the time: found days when the modification time of ALL datasets is reset ➔ all past datasets would be processed again
      - Possibility to take this info directly from the PS2 DB being explored
- The reverse task to find datasets declared "no longer good" needs discussions with the AMI team (this week)

Open-ended production

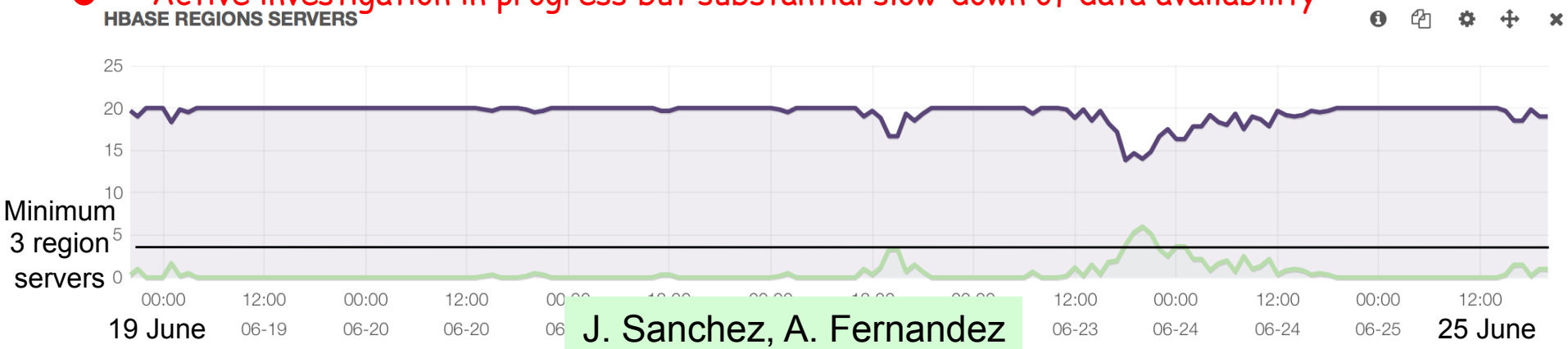
- Open-ended production id the ProdSys2 way to process datasets as soon as they appear
  - Used also by the derivation production
- We created the AMI tag for the EventIndexProducer transformation POOLtoEI_tf.py
  - AtlasProduction 20.1.5.4 and later contain EventIndexProducer-00-02-09 as AMI tag i2
  - EventIndexProducer-00-02-10 with fix for empty events will soon be in AMI tag i3
- We created 5 containers to be used for the open-ended production:
  - group.proj-evind:{data|mc}15.EventIndex.OpenEnded.{AOD|DAOD|EVNT}.i2
    - Same format as for derivation production (as it was last week)
- We simply add the list of datasets to be indexed to this container with Rucio
- ProdSys2 recognises the new datasets and defines tasks to run the EI producer
  - One task for each TID dataset
  - Group set to SOFT
  - Project set to {data|mc}_evind
    - Only used for output dataset with logfiles
  - Jobs set initially to run on 50 input files/job but we discovered now very large merged files (up to 7 GB/file) so set the max input site (15 GB/job) instead

✓ All OK so far
  - Apart from problems mentioned above, now all fixed

✓ We have a Production Manager team: A. Favareto, S. Gonzalez, F. Prokoshin

Data Validation

- Data validation is needed between the reception of data from the brokers and the insertion into Hadoop mapfiles and catalogue
  - Basic checks of consistency and completion of the data for each job and file
- Validation of Tier-0 data established since several months
- Validation of Grid-produced data in deployment
- BUT:
  - Validation uses HBase to temporarily store and sort the event information
    - One record per event
  - HBase is also used by other system components
  - There can be only one instance of HBase in a Hadoop cluster (in current configuration)
  - The insertion and sorting of received data by the validation process interferes negatively with data queries and retrieval
- Active investigation in progress but substantial slow-down of data availability

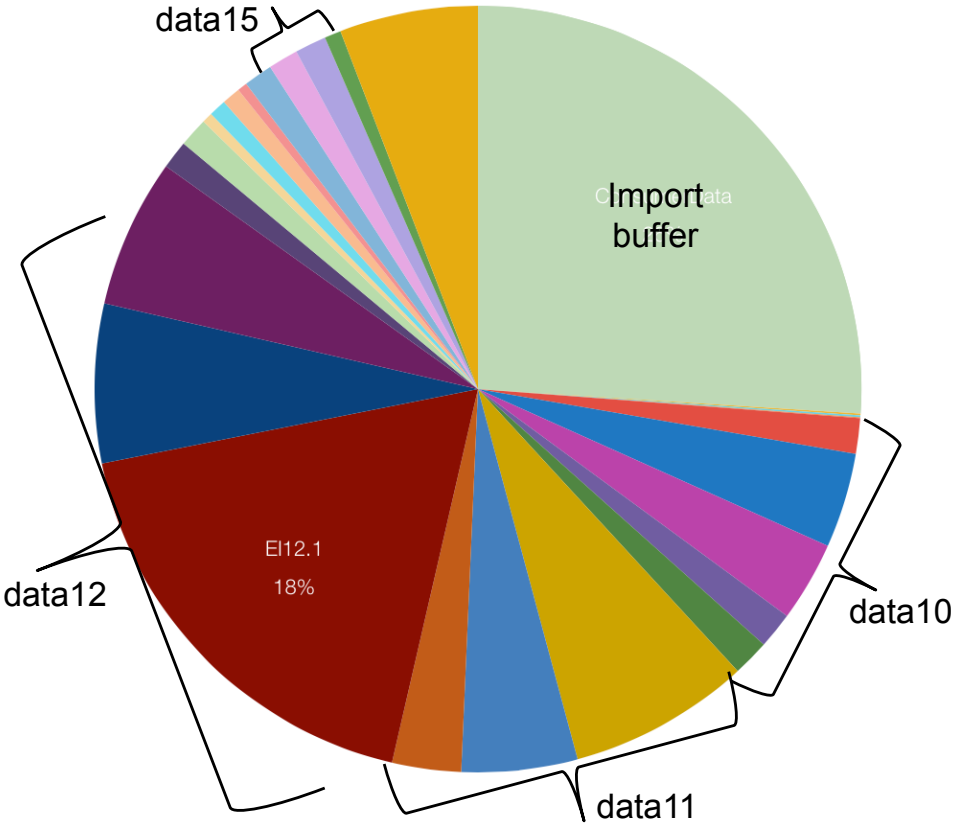


J. Sanchez, A. Fernandez

Hadoop Core

J. Hrivnac
R. Többicke
R. Yuan

- Successful validation of a dataset triggers
  - Data move to Hadoop mapfiles
  - Indexing and cataloguing
- ✓ All OK
- Data are then available for queries through CLI, python API or GUI



- About 40 TB of data in Hadoop now:
  - All Tier-0 production 2009-2015
    - Ran on AODs and collected pointers to RAW, ESD and AOD
  - In progress:
    - mc15 EVNT for Event Service
    - data15 AOD + DAOD from Grid production
- All OK so far but for validation troubles with HBase that slow down the throughput

Trigger decoding — Monitoring

- Trigger tables imported into Hadoop (HBase) through COMA
  - Now new structure that includes all runs since 2009
- All Run 1 tables available for EI since a long time
- Run 2 tables recently provided and now imported
  - Run 2 data need re-indexing for trigger
  - New runs will be uploaded daily
    - Before EI info is produced at Tier-0, validated and stored in EI

F. Prokoshin

- Monitoring in Kibana:
  - https://meter.cern.ch/public/_plugin/kibana/#/dashboard/temp/EventIndex::Availability

S. Cardenas
(next talk)

Event Picking

- Event picking from pathena should have been a no-brainer but turned out to be more complicated
  - Wrapper to provide the same interface as the old runEventLookup.py for the TAG DB turned out not to provide identical output to EI client
    - Fixed now but took some time to find the reasons
- Active investigation in progress (Andrea Favareto)
  - Found working combination of data and s/w releases for Run 1 and Run 2
  - Checking that it works for all formats and options available in pathena
  - Updating documentation and tutorial
- In addition we are preparing the option to give to pathena a complete EI query instead of the run-event list
  - Pathena would call runEI.py and get back the list of GUIDs (like for traditional event picking).
  - Pathena would then return the selected events to the user.
  - Useful for small but automatic event selections based (for example) on rare trigger combinations.

J. Hrivnac, A. Favareto

Duplicate event detection

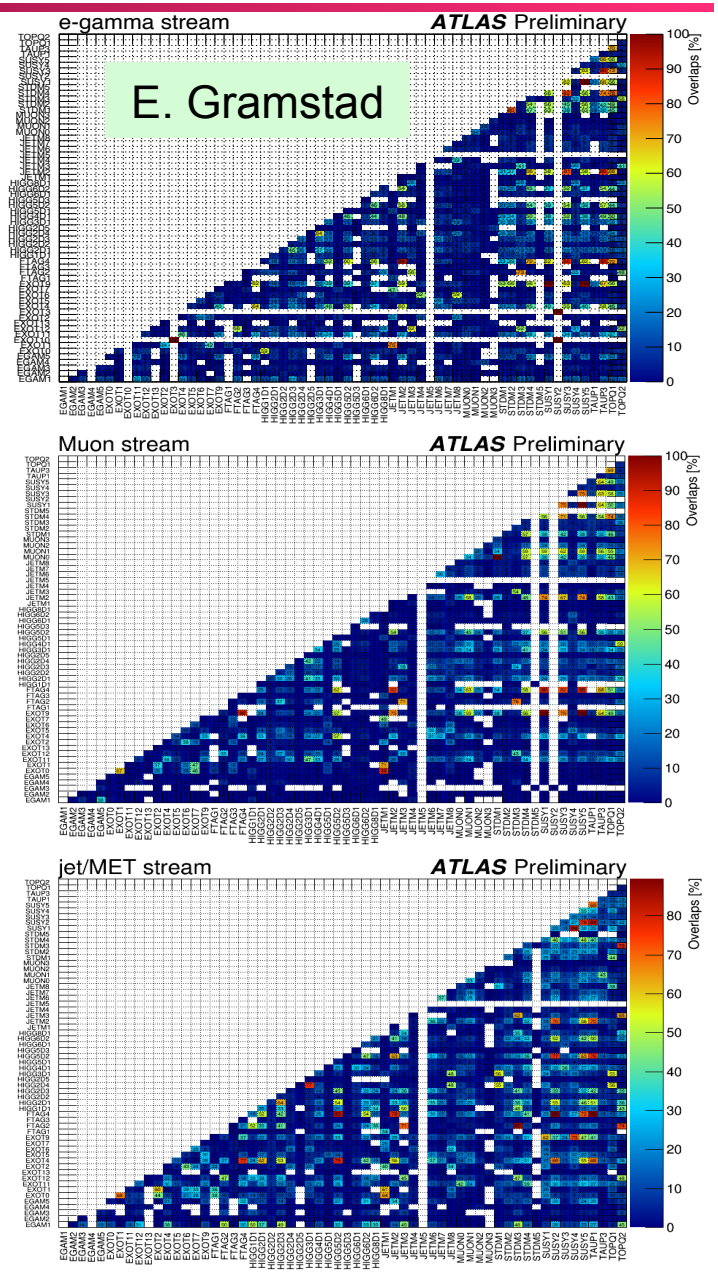
- Version of ATLAS code up to
  - AthenaMP-01-02-30-02
  - AthenaInterprocess-00-02-08

  had a bug that could show up if multiple athenaMP jobs started at the same time on the same worker node
  - Events could be mixed between the jobs
  - Some events could be processed multiple times and some not at all
- This problem is fixed in AtlasProduction 20.1.5.9 and the corresponding simulation release
- Data produced with the buggy version (mostly mc15) must be cleaned
- The validation process in EventIndex counts events and sorts them by run-event number, so duplicates are visible
  - The discovery of affected files and datasets processed through EventIndex internal validation can be done routinely
    - Done now for existing mc15 AOD and DAOD

J. Sanchez, A. Fernandez

DAOD overlaps

- The Derivation Framework group asked us to provide a tool to find the fraction of overlapping events between derivation streams
- A first test was run with reprocessed data12 but it's no longer very useful as the trigger streams changed for data15
  - Overlap matrices were produced by hand while waiting for a more automatic system (some problems solved in the meantime)
  - Anyway here they are (as examples):
- A second one with early data15 runs is in preparation
  - The idea is to have this tool running all the time in the background and to check the results from time to time or when derivation streams change



J. Sanchez, J. Cranshaw

Conclusions and Outlook

- All pieces are in place
- Major problem is HBase overload
  - Very active investigations
  - Also search for alternative solutions for part of the workload
- "Last mile" tools coming together and working
- Automation in progress
  - Hope to achieve "shifter level" operation stability in a few months
  - But need more intensive data throughput to really test robustness
- Further performance improvements under discussion
- Finally:
  - Not too bad considering only 4 FTEs and 2.5 years!