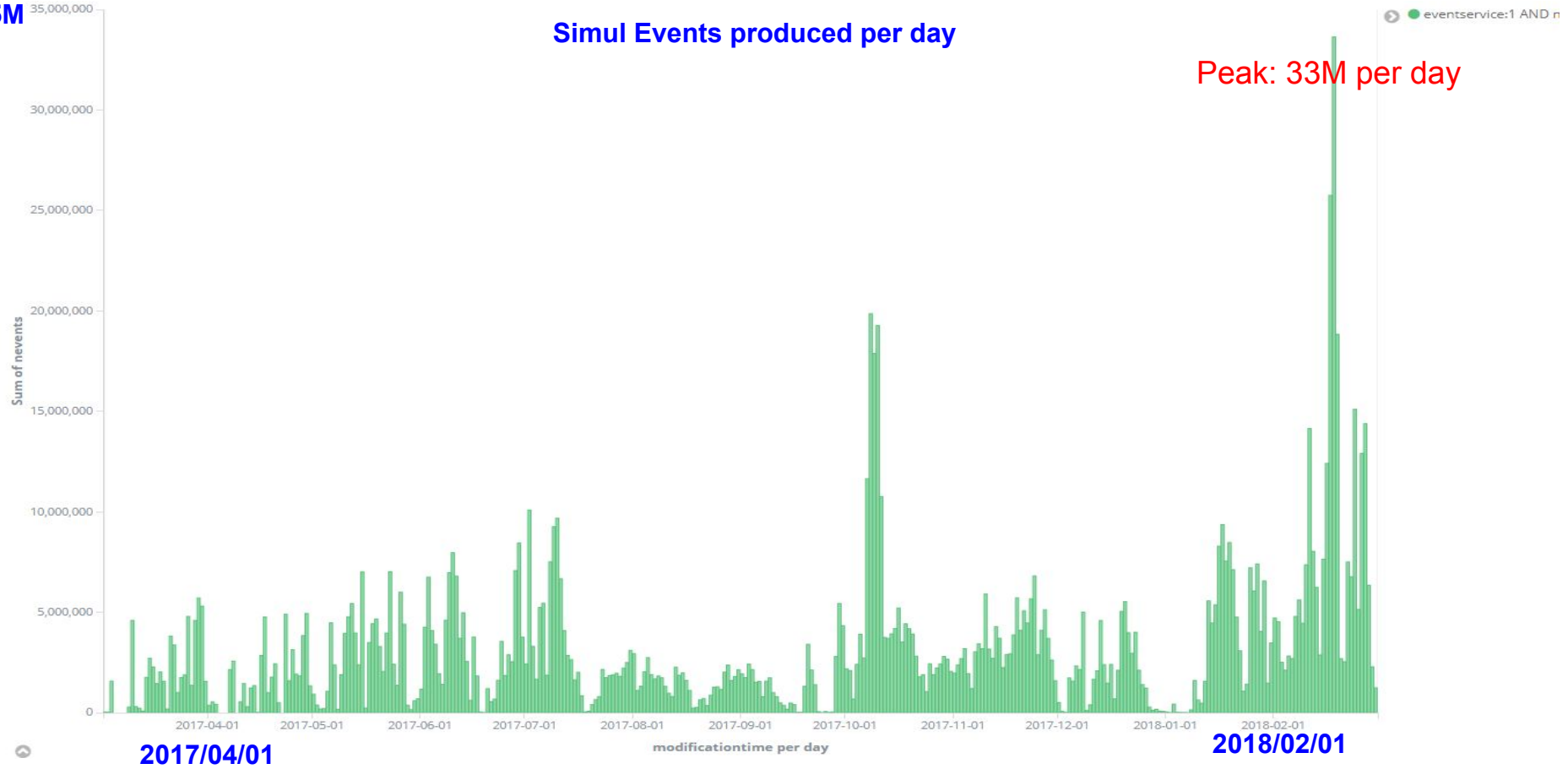
EventService for sites

Wen Guan on behalf of EventService team

Events produced per day (one year)

35M

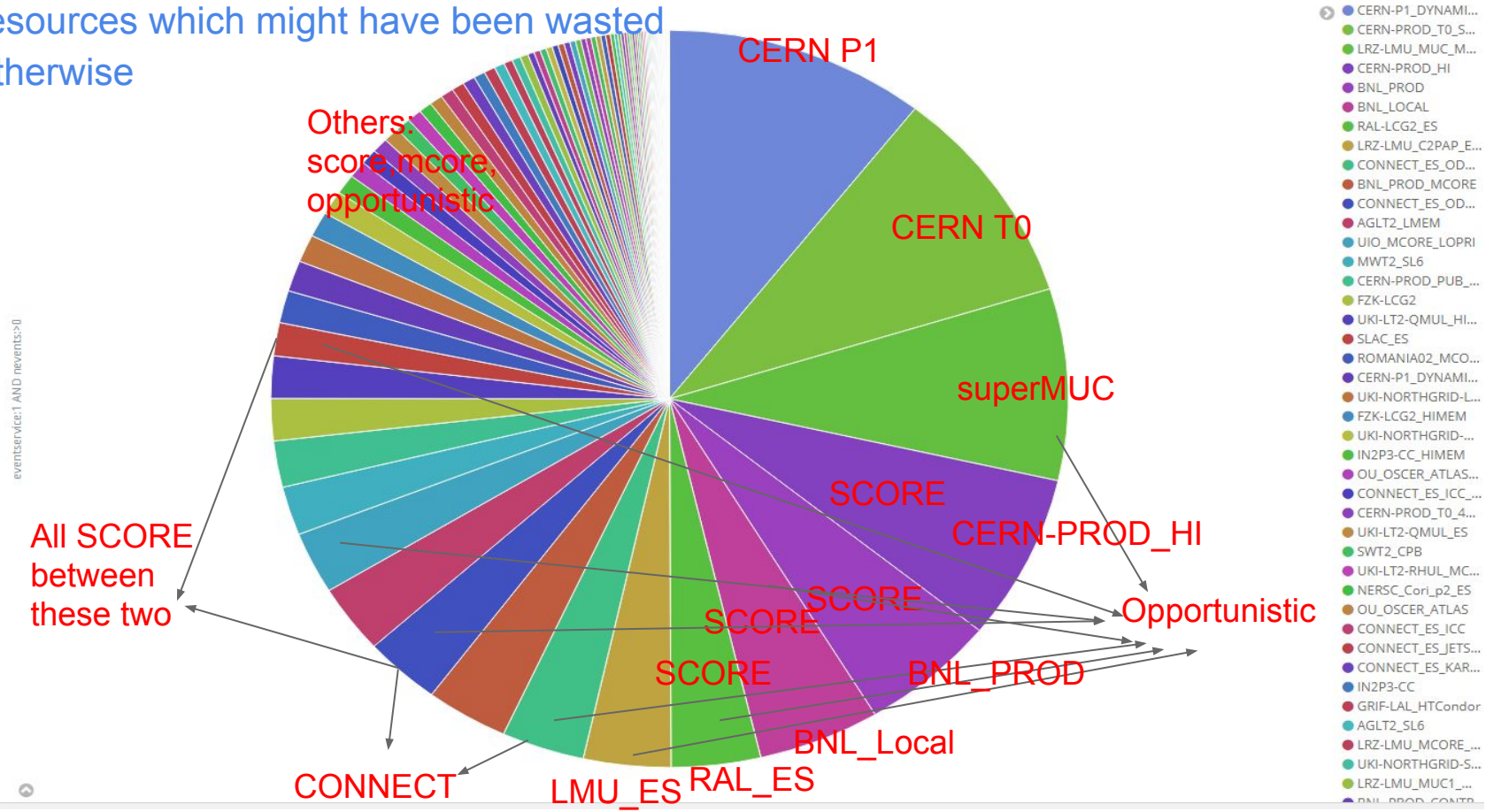


2017/04/01

2018/02/01

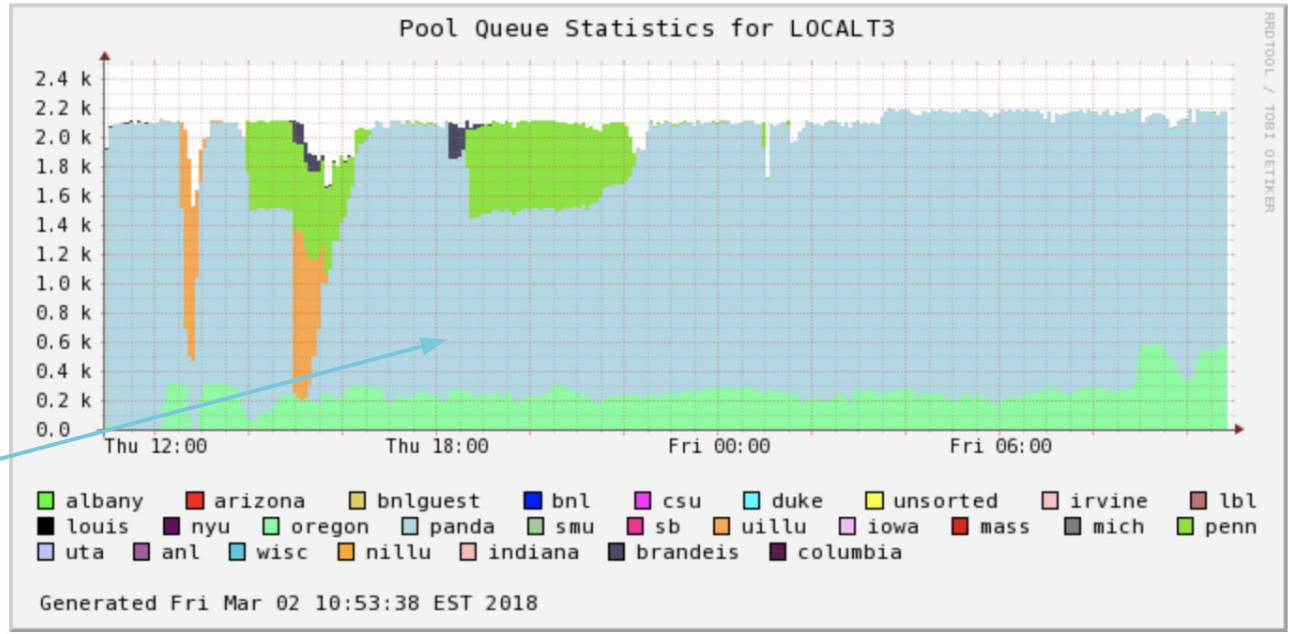
Events produced per site (one year)

ES grabs a lot of SCORE and opportunistic resources which might have been wasted otherwise



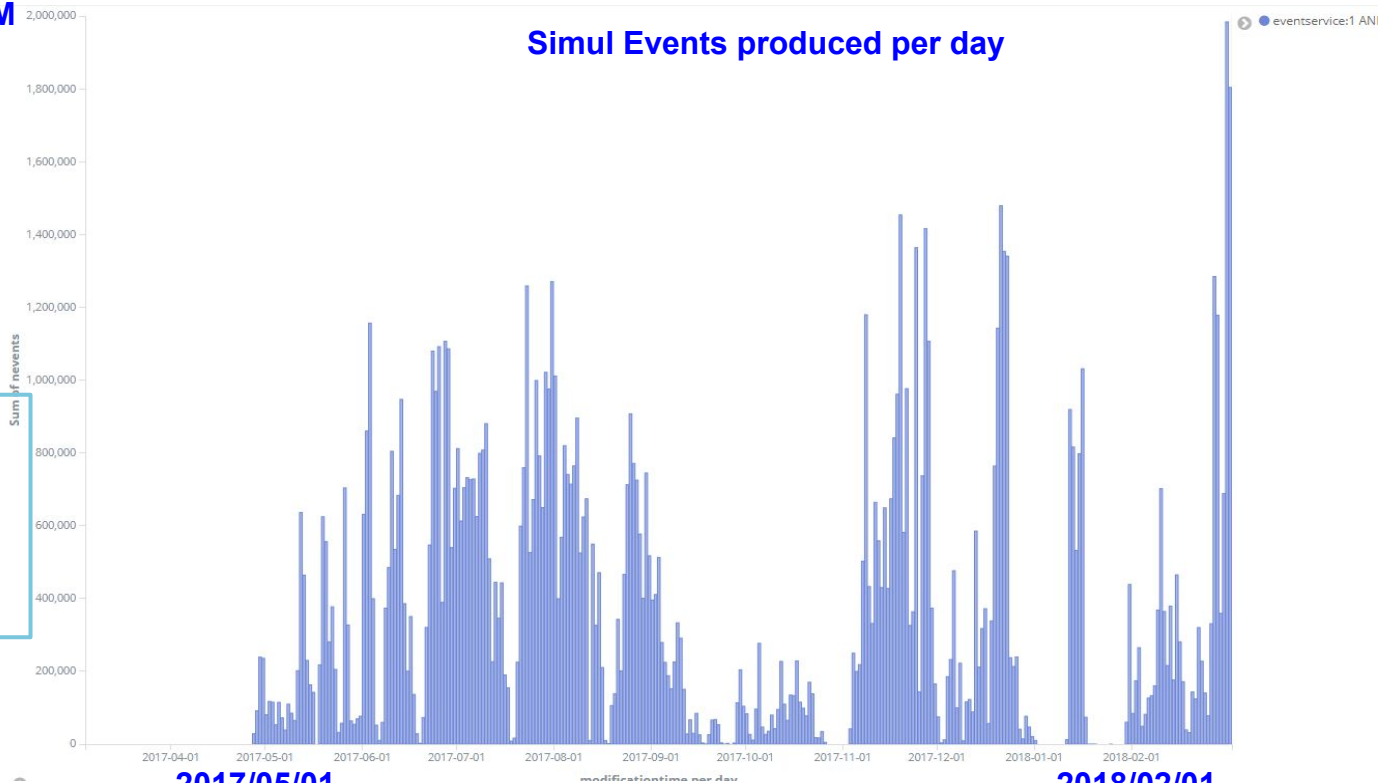
Opportunistic resource: BNL_LOCAL(SCORE)

EventService used it to produce a lot of events when the cpu is not used by local users.



Opportunistic resource: CONNECT* (one year)

2M



EventService has produced a lot of events by using CONNECT* opportunistic cpus.

The storage is much stabler

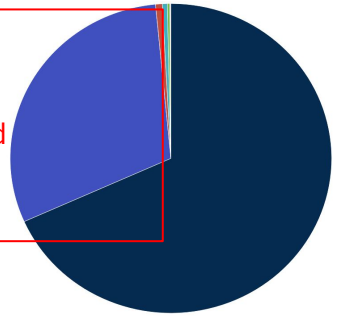
Few failures: counted every 3 hours

- Much stable now
  - Few failures to stageout and stagein

EventService

ES failure reason:

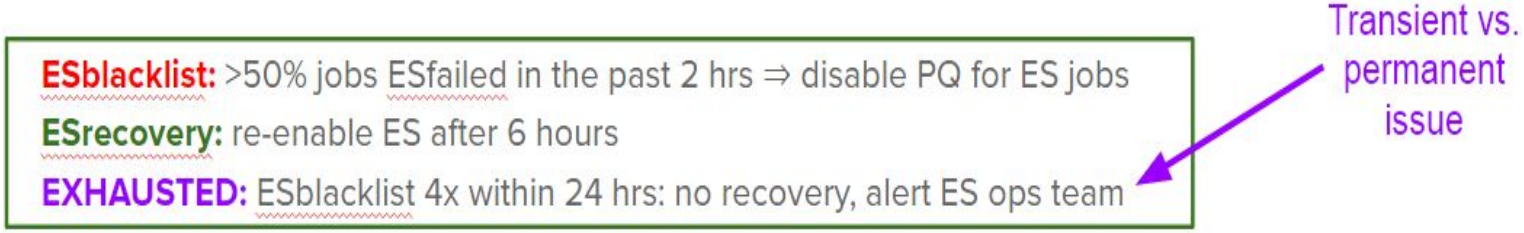
- 68% es_merge_failed
- 30% pilot_failed
- 2% other



- Storage
  - For long time we were focusing on this part. It's much stable now.
  - Next step:
    - Two copies of pre-merge files to solve the stagein problem for es-merge jobs
      - (last year we found a big part of ES failures is caused by es-merge jobs)
      - one to OS and the other to OS or DATADISK
      - one is periodically (currently using) and the other is done at the end of the job(for opportunistic queue, we cannot make sure we can do a copy at the end of the job; we may use an easy option: using a different time period. For example, the first copy is 10 minutes period, the second copy is 1 hour or even longer).
    - Es merge job will automatically select the other copy if it fails to stagein one copy.
    - Remote Stagein and ESS (can be discussed in the splinter meeting)
      - ES resources are not predictable (can appear and disappear at any time)
      - The resource can disappear when waiting to brokerage data
      - Subscribe the data and release jobs immediately without waiting the data is ready
        - When panda detects some sites run out of jobs (panda knows pilot getJob with no jobs).
        - Will solve the problem like that BNL run out of jobs.
        - Maybe can send the data to rucio cache for all sites in one cloud?
      - Allow pilot to stagein remote close replica in this case

HammerCloud blacklist sites

- ESblacklist: ESfailures ⇒ disable ES for the resource (already existed)



- New/Will come blacklist rules
  - Preempted too frequently: A lot of preempted jobs with walltime less than 10 minutes
    - Will not produce any events but increase the retry counter of events
  -

EventService commissioning

- After ES storage was much stabler, we tried to move on to scale, two issues slowed down us
  - Brokerage/share/priority issue when scaling (next slide)
    - Some sites get jobs but it cannot run because of opportunistic or priority/share
    - Some sites have free resources but it cannot get jobs (total throttler: should not be too high than the total running jobs)
  - Site issues (next next slide)
    - Lots of failures at some sites will fail consumers, jobsets, and tasks
- We slow down to focus on these issues

EventService brokerage/priority/share when scaling

- Scale to normal PQ
  - We tried to scale many normal PQs to run ES (jobseed=all)
    - Idea: When it run out of normal jobs, run ES
  - Frequently ES are scheduled to these PQs but cannot run
    - because ES tasks' priority is much lower than ordinary MC and PQs are busy with ordinary MC
    - Opportunistic resources can disappear at any time
- Scheduled but not run jobs contribute the throttler
  - Throttler to control total released jobs to avoid too many released jobs comparing available running cpus.
  - Some queues which can run jobs cannot get enough jobs because we reached the throttler
- Scheduled but not run jobs increase the retry counter
- Opportunistic and other resources for ES are not predictable
- Will improve this part
  - Using different share for ES?
  - Not scheduled ES to PQs which have high priority ordinary MC?
  - Schedule and release ES jobs immediately when panda receives "getJob" calls from pilot but no available jobs? (with remote stagein enabled)

EventService sites when scaling

- Old saying:
  - If the site cannot run ordinary jobs, we can try ES on it.
  - As a result:
    - A lot of failed jobs because of problems other than preemption
- ES cannot solve problems other than preemption currently.
- Policy we would like to address:
  - Blacklist sites which have a lot of failures (even it has a part of successes)
    - Normally preemption in ES is not marked as a failure for preemptable Panda Queues (Pledgedcpu=-1).
    - Failures other than preemption, such as stagein/stageout, athenaMP condition db, …...
  - Blacklist sites which produced very few events per job and with a lot of closed jobs
    - A lot of jobs are preempted in few minutes (less than 10 minutes, not producing events)
    - With only few jobs producing few events
  - If ES only runs on sites passed HammerCloud simu tests, it will be very good to avoid not-validated sites to join the production workflow
    - ES currently at first is a simulation job, it's the same AthenaMP with special event range channel.
      - It requires the site can run normal AthenaMP.
    - HC simu test only contains 3 events
      - If the site cannot pass it, it's not good to include it to ES
    - HC can avoid many errors such as stagein, condition db errors and other AthenaMP errors, which are the most frequent failures (they are not ES related)
    - For preemptable queues, maybe not mark a preempted job as a failure in HC
  - For problem sites, we can define some test ES jobs to debug (set site BROKEROFF), but should not use production jobs to do heavily tests (should not cause a lot production failures)

Future

- Scale to more PQs
- Looking for more SCORE and opportunistic queues
  - T3 as opportunistic resources
    - Local users can preempt ES jobs
    - When local users are not using the resources, ES can use them