A scalable monitoring for the CMS Filter Farm based on elasticsearch

Srećko Morović,
Salvatore Zaza
CERN

On behalf of the CMS DAQ Group

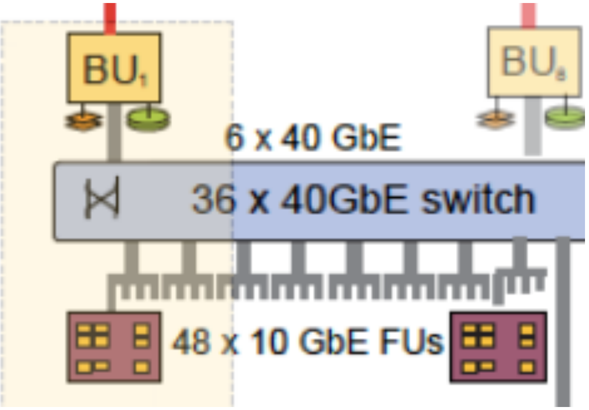
21st International Conference on Computing in High Energy and Nuclear Physics (CHEP 2015) Okinawa, Japan, April 13-17, 2015

Introduction

* DAQ2 Event Builder, Filter Farm and Storage System described in talks:
E. Meschi : File-based data flow in the CMS Filter Farm
R. Mommsen: A new Event Builder for CMS Run II
L. Darlea: Online data handling and storage at the CMS experiment

CMS DAQ2 Filter Farm

- Divided into 62 sub-farms (“appliances”)
- Fully built events copied to Filter Units (FUs), up to 20 FUs / appliance
- 40-Gbit network (BU) to 10 Gbit (FU) per appliance



File-based Filter Farm (FFF) monitoring

- JSON metadata produced at each FFF data handling stage, accompanying and describing:
  - RAW data in ramdisk
  - FU HLT (accepted) output per stream
  - Three stages of output file merging (micro, mini, macro merging)
- Of interest for monitoring: event rate, completeness of processing/merging per lumisection (LS), CPU usage statistics, disk occupancy, logs

Requirements on the monitoring system:

- Scalability – scaling to large number of HLT nodes
- Low latency – fast feedback for the Online shift crew/experts
- Ability to directly insert JSON documents

Elasticsearch

- NoSQL DB and search server based on Apache Lucene library
  - Open source, large user base and community, used commercially
- Quasi-realtime
    - Low latency document insertion and searching
- Indexed JSON document storage using lightweight schema
- RESTful JSON-based HTTP interface
    - Multitenant (serving multiple clients)
    - Several client libraries available
- Clustered architecture
  - Data distributed over *indices* which can be split over many data nodes in the cluster
- Documents searched across cluster with a single query
  - Initiated from any node in the ES cluster
  - Possible aggregation according to algorithms
    - e.g. histogram documents with per-stream bins, or time interval bins

FFF elastisearch cluster

- Dedicated cluster of ~ 20 machines running elasticsearch instances (**Central ES**)
- Permanently storing low-volume information aggregated from the Filter Farm appliances:
  - Run information
  - Filter Farm cluster status
  - Logs
- Separate set of indices used for Central DAQ, and for other dedicated DAQ clusters (miniDAQ, DAQVal …)
- Example: “end of luminosity” document injected by one of the BUs:

```
{
  “fm_date": "2015-01-09T03:34:17.839651”,
  "TotalEvents": 233528,
  "NEvents": 10368,
  "ls": 1,
  "NFiles": 30,
  "id": "run231913_ls0001_EoLS_bu-c2f16-19-01”
}
```

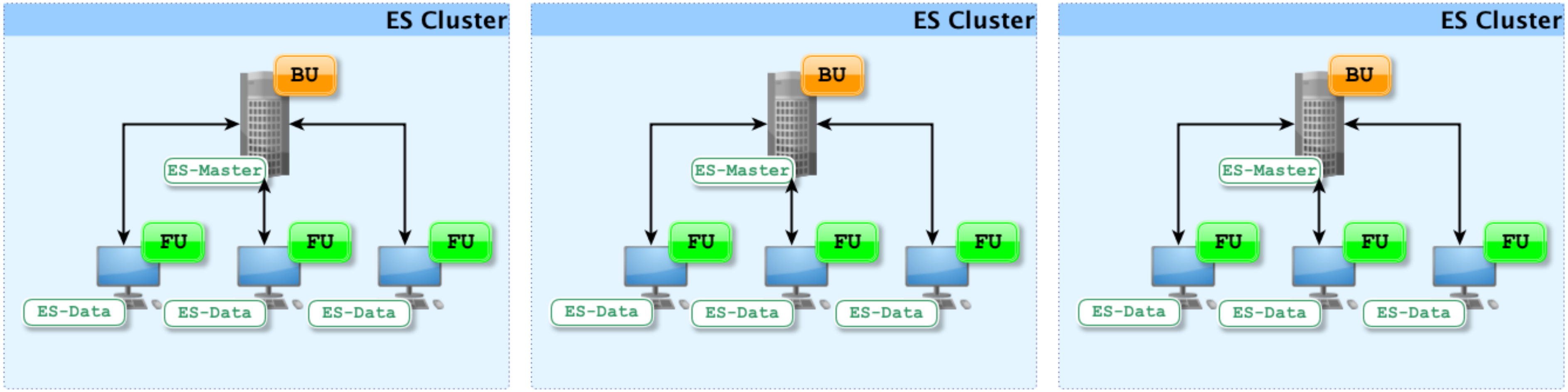
Contains number of events and files in LS 1 in one appliance

Retrieved by query:

```
curl –XGET http://es-srv:9200/run_index/eols/_search –d’{
"query":
          {"prefix”: { "id":"run231913"} },
"query":
          {"term": {"ls":1702} }
}’
```

FFF appliance ES clusters

Appliance 1 Appliance 2 .... Appliance N

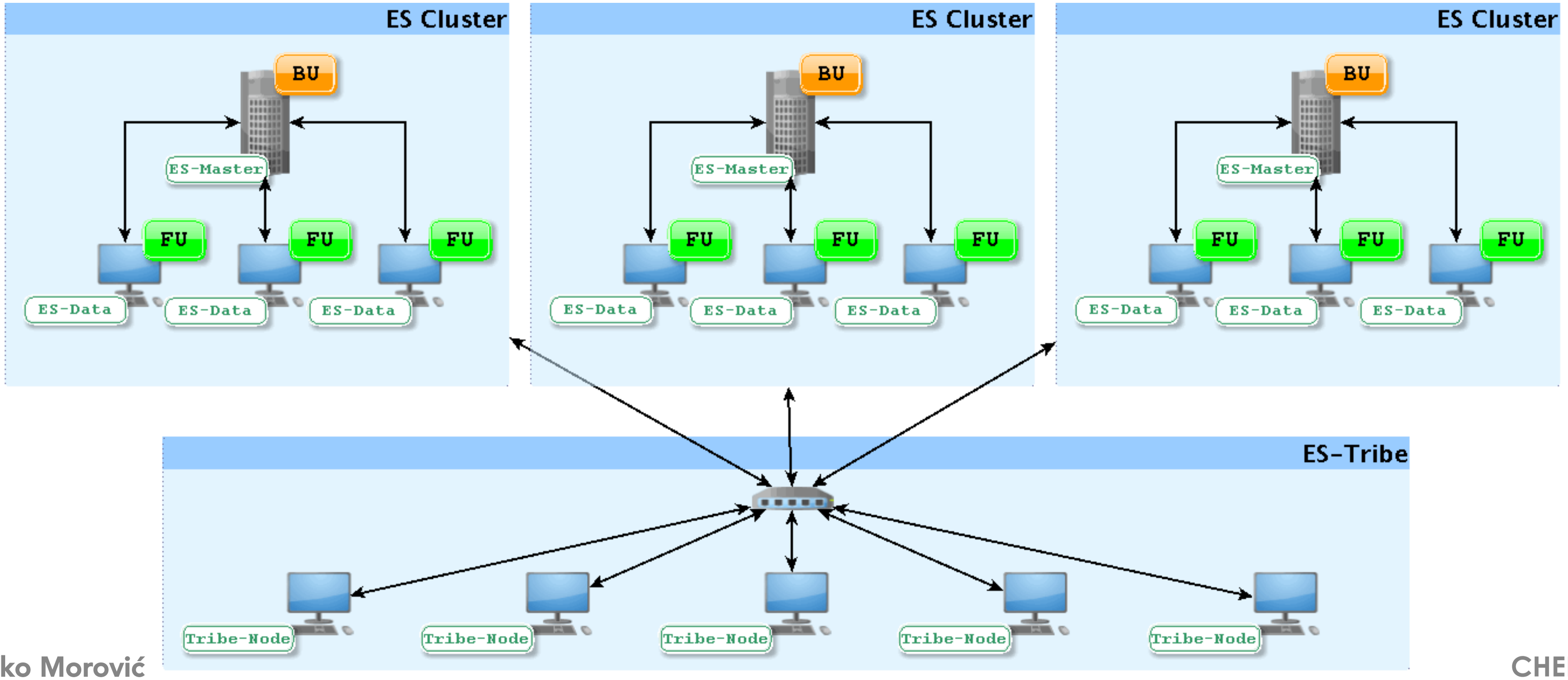


- Elasticsearch service running on each FFF BU and FU node
  - Profit from FFF architecture: ES clusters match appliance (unit of processing) structure
  - lightweight: low CPU (~ 10% single-CPU) and moderate memory usage (≤1 GB JVM heap)
- Natural horizontal scaling in FFF
  - processing nodes are elasticserch data nodes
  - Appliances clusters are independent of each other
- FFF services on FUs inject JSON files into local ES instance as they appear
  - Storing "full-detail" run information (down to process-level granularity)

Connecting clusters

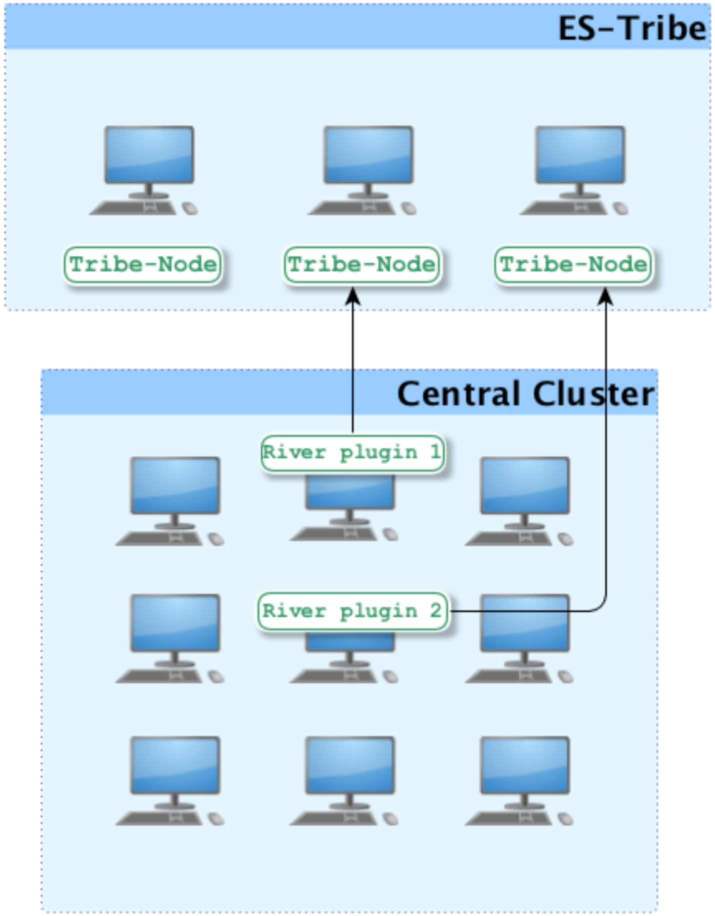
- Need to collect monitoring data from appliances
  - Running queries on each cluster individually is cumbersome
  - Requires sending query to each cluster and client-side merging JSON results

- Using ES **Tribe** feature – allows to form cluster of clusters
  - Tribe nodes join each cluster as non-data members
  - Queries to tribe nodes run on all connected appliance ES clusters
  - ~ 10 (independent) tribe nodes for redundancy and load balancing



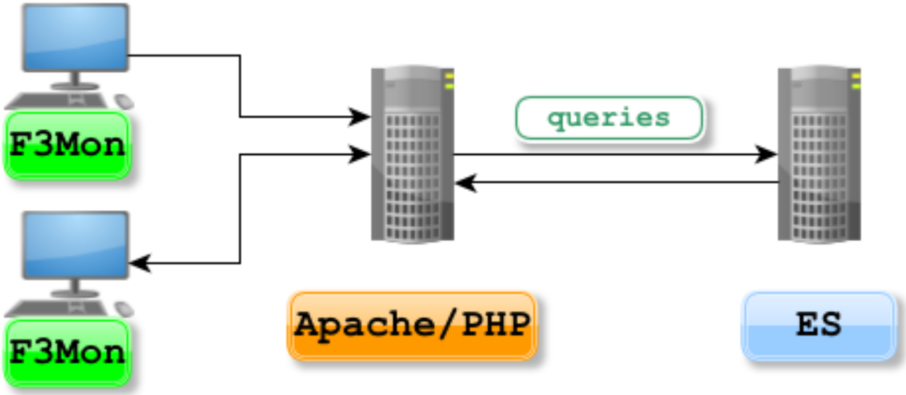
Data collection from appliances

- Some information written to central ES directly (e.g. logs, ramdisk JSONs)
- FU-level run data aggregated into central ES by querying appliances
- RunRiver data collector plugin
  - built using ES plugin API
  - Plugin automatically started and managed in one of cluster nodes
  - Independent instance per run
- Execute queries in Tribe and store aggregated results in central ES cluster
  - Special aggregation queries result in compact histogram-like documents, small enough for handling by the monitoring web UI



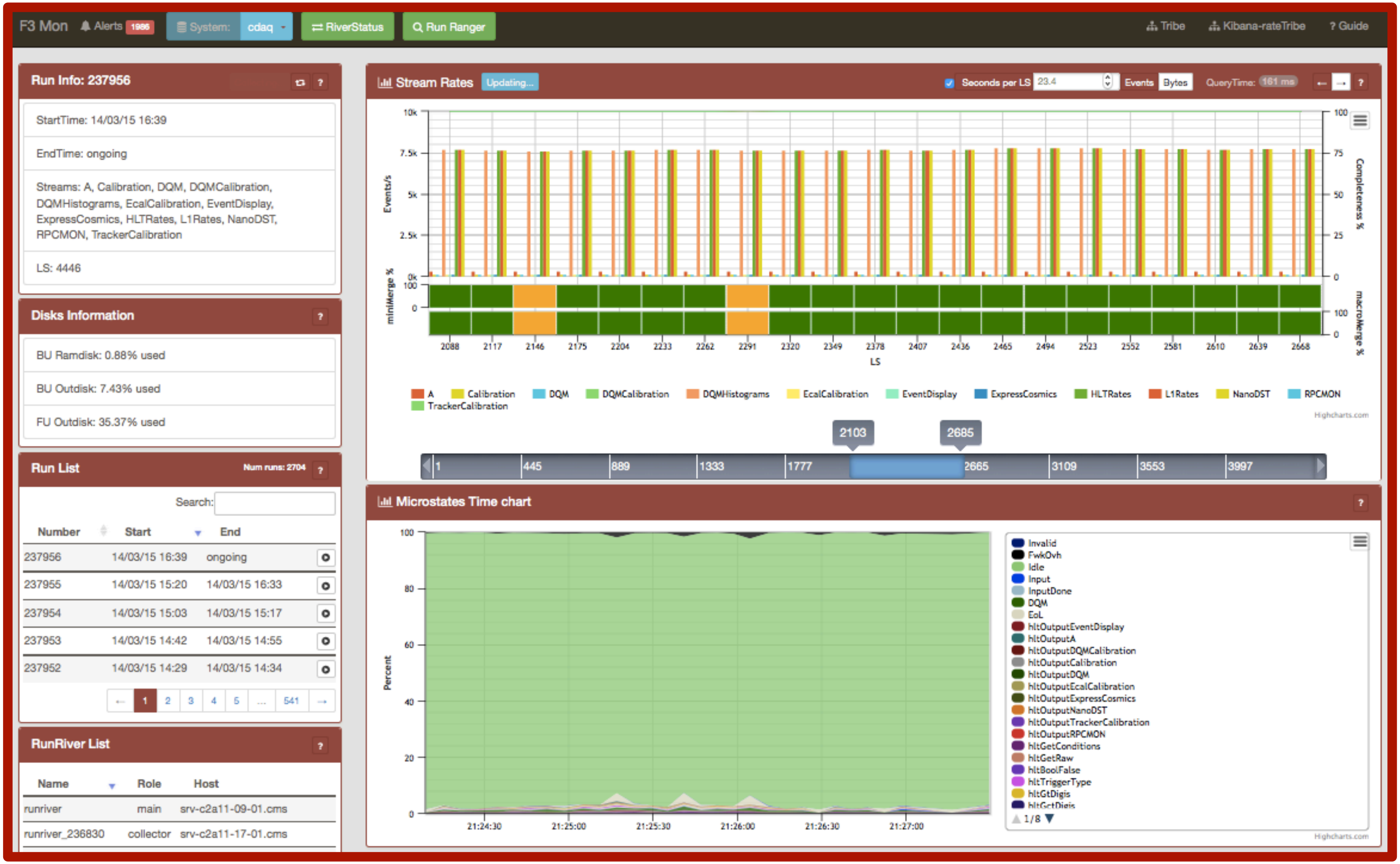
F^{3}Mon GUI

- FFF web-based monitoring built on top of elasticsearch infrastructure
- One of general DAQ monitoring pages used by everyone
- Interface based on modern JS libraries
  - Bootstrap, jQuery, HighCharts
- Server-side PHP fetches data from central ES



  - Multiple server instances, one running on each central ES host
  - Central ES is isolated by direct access by clients (better data security)
  - Possible to use server-side caching for expensive queries
- PHP runs queries on request from F3Mon, which is polling for new data
  - Parameters Determined by what the user has requested (e.g. run number)

F^{3}Mon GUI



Run Info: 237956

StartTime: 14/03/15 16:39

EndTime: ongoing

Streams: A, Calibration, DQM, DQMCalibration, DQMHistograms, EcalCalibration, EventDisplay, ExpressCosmics, HLTRates, L1Rates, NanoDST, RPCMON, TrackerCalibration

LS: 4446

Disks Information

BU Ramdisk: 0.88% used

BU Outdisk: 7.43% used

FU Outdisk: 35.37% used

Run List Num runs: 2704

Search:

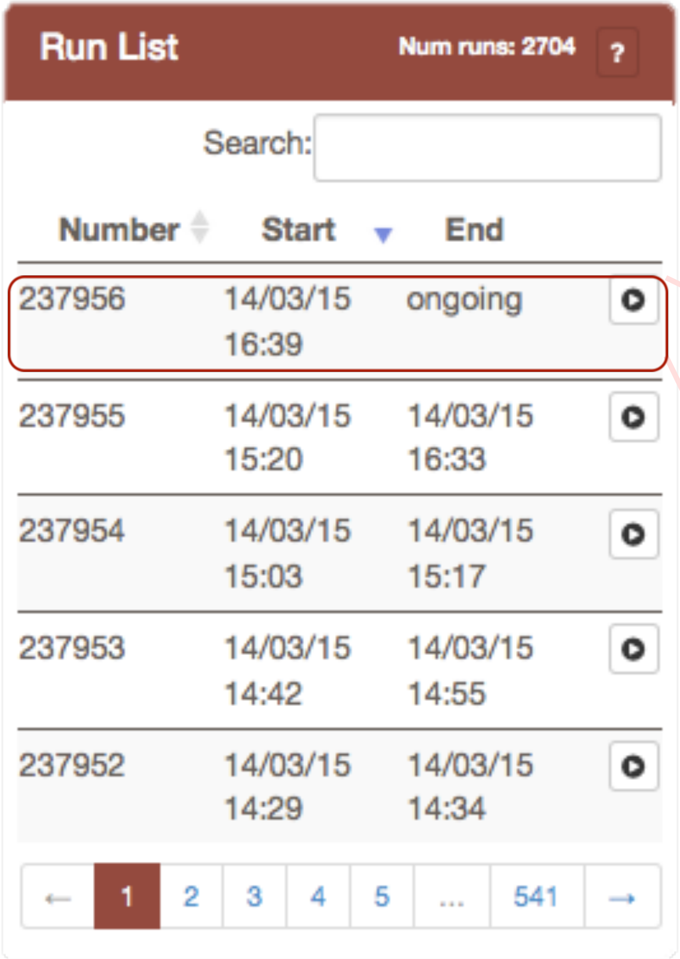
Number	Start	End
237956	14/03/15 16:39	ongoing
237955	14/03/15 15:20	14/03/15 16:33
237954	14/03/15 15:03	14/03/15 15:17
237953	14/03/15 14:42	14/03/15 14:55
237952	14/03/15 14:29	14/03/15 14:34

← 1 2 3 4 5 ... 541 →

RunRiver List

Name	Role	Host
runriver	main	srv-c2a11-09-01.cms
runriver_236830	collector	srv-c2a11-17-01.cms

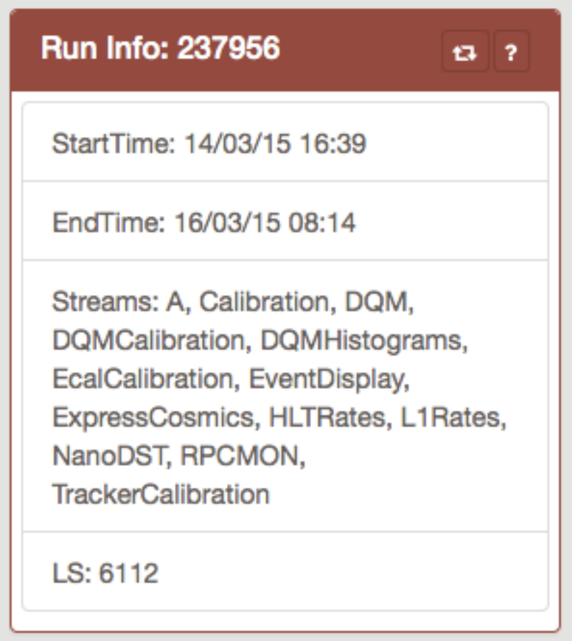
Run information



Run List Num runs: 2704

Number	Start	End
237956	14/03/15 16:39	ongoing
237955	14/03/15 15:20	14/03/15 16:33
237954	14/03/15 15:03	14/03/15 15:17
237953	14/03/15 14:42	14/03/15 14:55
237952	14/03/15 14:29	14/03/15 14:34

- By default, UI shows status of the live run
- Full history of runs with all information available for selection within the UI



Run Info: 237956

StartTime: 14/03/15 16:39

EndTime: 16/03/15 08:14

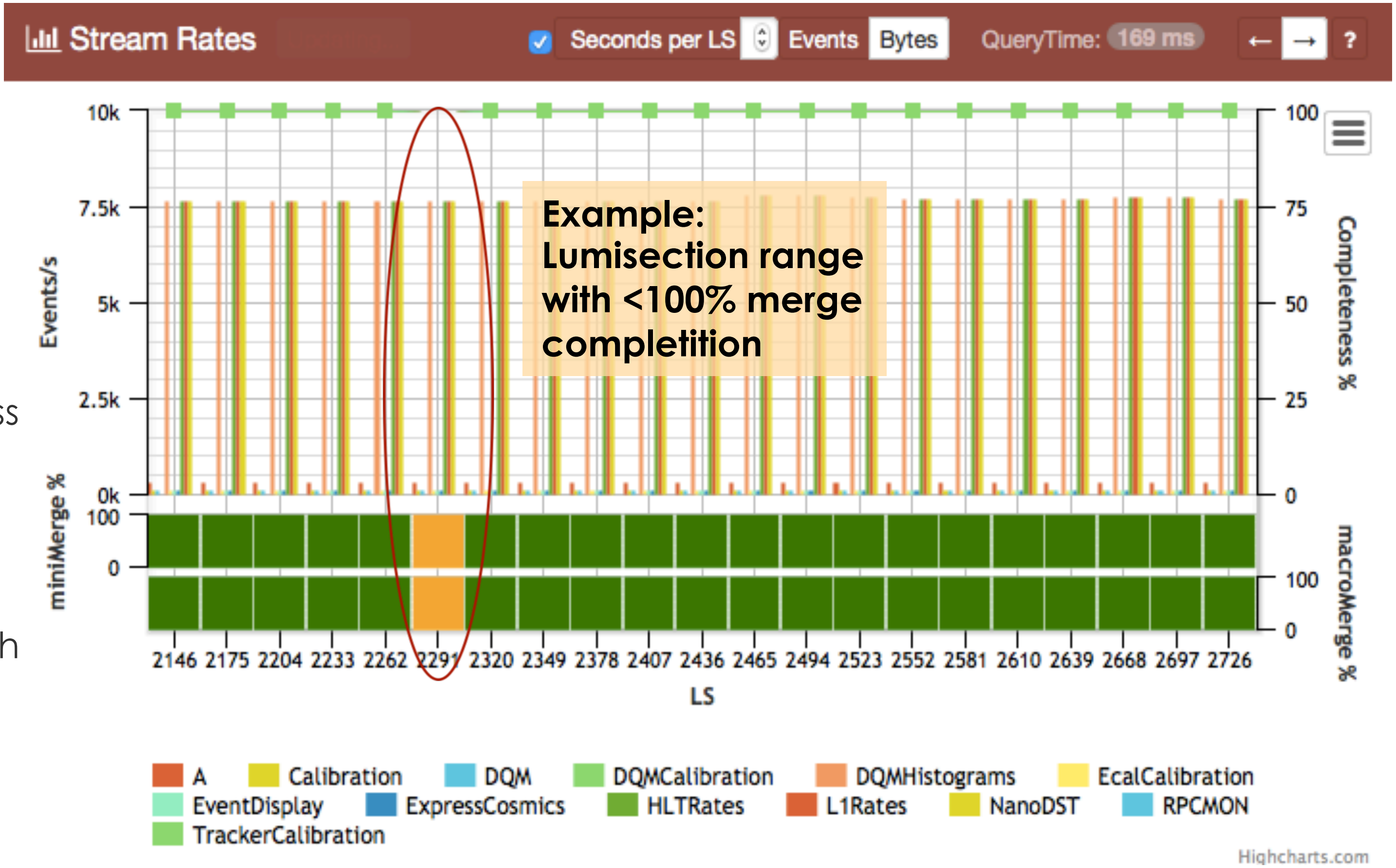
Streams: A, Calibration, DQM, DQMCalibration, DQMHistograms, EcalCalibration, EventDisplay, ExpressCosmics, HLTRates, L1Rates, NanoDST, RPCMON, TrackerCalibration

LS: 6112

HLT output/merger monitoring

- Display of aggregated per-LS HLT-output rates and merging completion for each merging stage

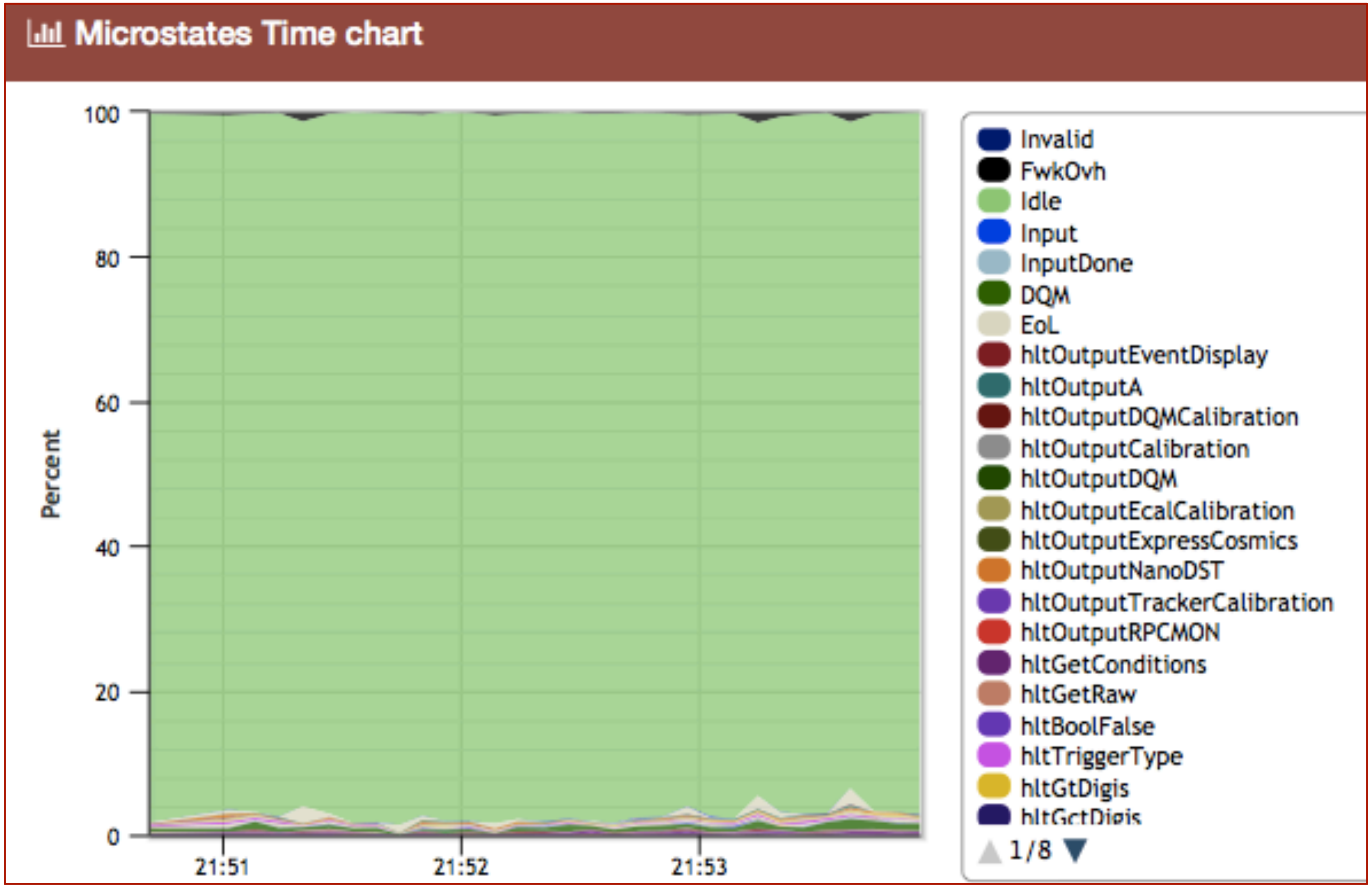
- By default: displays status of current (recent) lumisections
  - Ability to display any lumisection range in the run
- "drill-down" (on-click) available to display merging completeness per stream and appliance level
- Example: LS range with incomplete merging problem



Example: Lumisection range with <100% merge completition

HLT CPU usage monitoring

- Aggregated process-level CPU usage
- Breakdown per HLT module (important to identify inefficiencies in HLT)



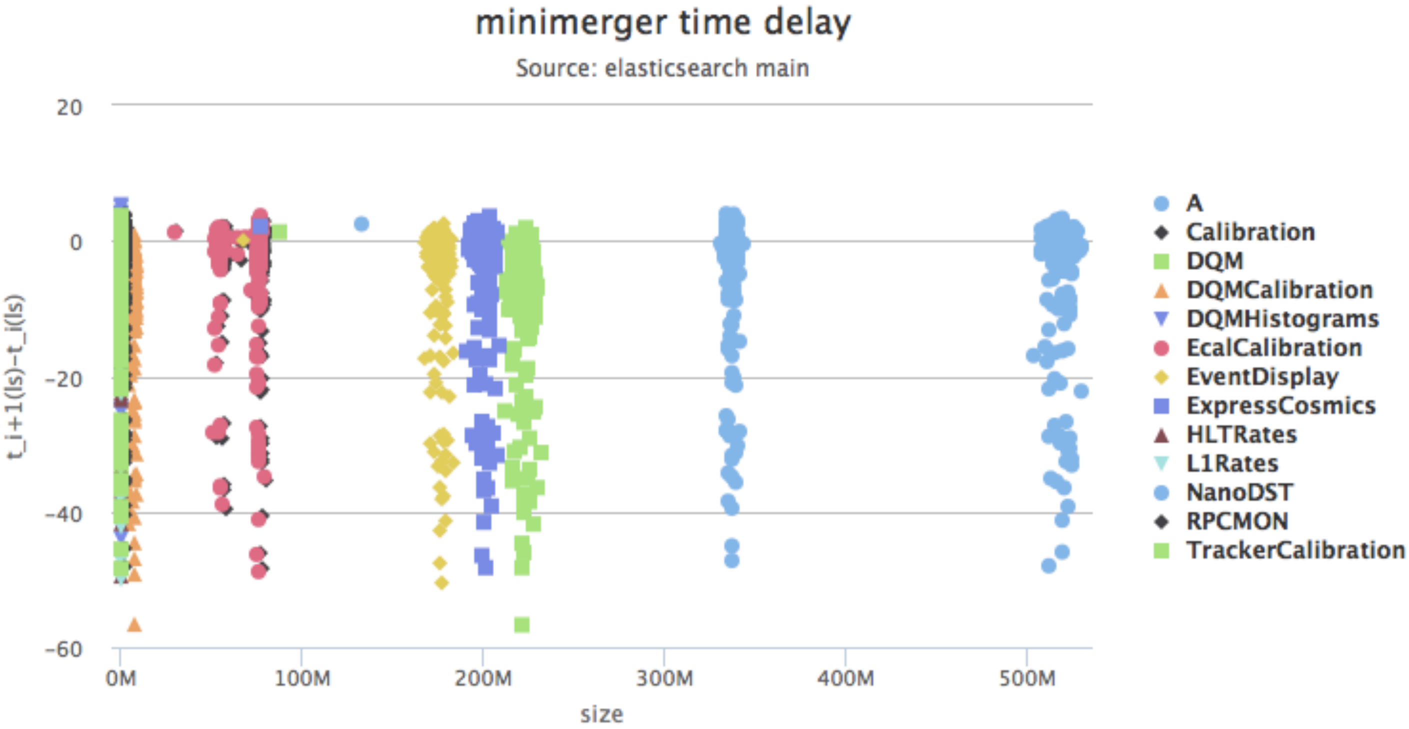
Example: period with the Filter Farm >95% ide

Expert tools

- es allows cross-correlation of monitored variables without a need for a-priori adjustment on how data is stored/indexed

  (possible to analyze data in a way that was not originally intended)

Example: a tool showing correlation of merging latency vs. file size per stream



minimerger time delay

Source: elasticsearch main

Aggregation query:

```
{
  "query": {"term":
    {"_parent":237955} },
  "sort":{"ls":"asc"},
  "aggs":
    {"streams":
      {"terms":
        {"field":"stream","size":100,
         "order":{"_term":"asc"}},
         "aggs":{
           "lss":{
             "terms":{
               "field":"ls","size":100000,
               "order" :
                 { "_term" : "asc" }
             },
             "aggs":{"sizes":
               {"avg":{"field":"filesize"}
}}}}}}}
```

Summary

CMS Filter Farm monitoring has been implemented using elasticsearch

- Dedicated ES cluster for permanently storaging monitoring data
- Appliance clusters: ES running all BU and FU nodes,
  in the background of event processing

  →Horizontal scalability with Filter Farm expansion

- No big problems
  - Stable, used in production Central DAQ
  - Scales well, after solving several limitation issues (e.g. – had to increased threads limit in Tribe hosts)
- F^{3}Mon UI
  - Monitoring runs in Filter Farm (HLT processing, merging) with low latency
- Several expert tools built on top of the ES infrastructure
- Serving also non-web based clients (LabView)

Outlook

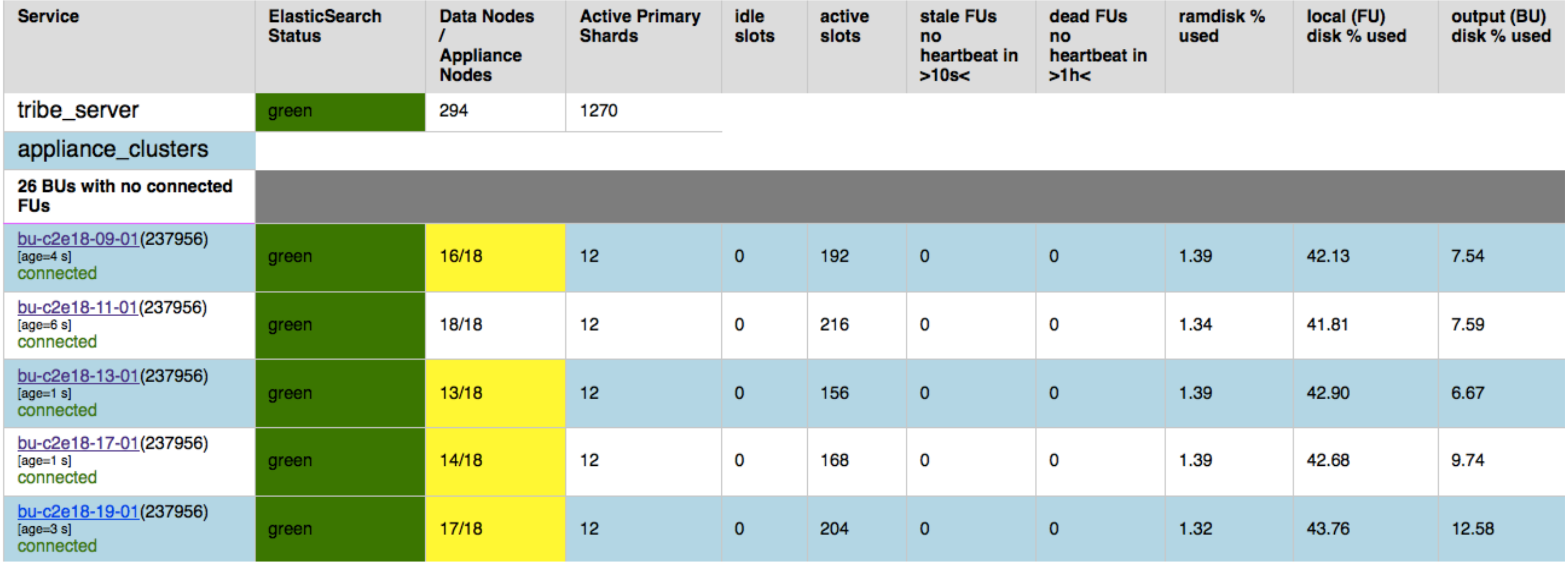
- Looking forward to integrate the full Filter Farm (nearly 50% complete)
- Experimenting with elasticsearch in other parts of DAQ

BACKUP

Filter Farm status page

Status of various parameters of the farm

- Used/Available CPU resources, ramdisk and disk usage, ongoing runs in HLT/ FFF, detection of problems (network problems, crashes) …



Service	ElasticSearch Status	Data Nodes / Appliance Nodes	Active Primary Shards	idle slots	active slots	stale FUs no heartbeat in >10s<	dead FUs no heartbeat in >1h<	ramdisk % used	local (FU) disk % used	output (BU) disk % used
tribe_server	green	294	1270							
appliance_clusters										
26 BUs with no connected FUs										
bu-c2e18-09-01(237956) [age=4 s] connected	green	16/18	12	0	192	0	0	1.39	42.13	7.54
bu-c2e18-11-01(237956) [age=6 s] connected	green	18/18	12	0	216	0	0	1.34	41.81	7.59
bu-c2e18-13-01(237956) [age=1 s] connected	green	13/18	12	0	156	0	0	1.39	42.90	6.67
bu-c2e18-17-01(237956) [age=1 s] connected	green	14/18	12	0	168	0	0	1.39	42.68	9.74
bu-c2e18-19-01(237956) [age=3 s] connected	green	17/18	12	0	204	0	0	1.32	43.76	12.58

Example: data injection

- End of LS document insertion:

```
curl –XPUT es-srv:9200/run_index –d'{
  "fm_date": "2015-01-09T03:34:17.839651",
   "TotalEvents": 233528,
   "NEvents": 10368,
   "ls": 1,
   "NFiles": 30,
   "id": "run231913_ls0001_EoLS_bu-c2f16-19-01"
}'
```

```
curl –XPUT es-srv:9200/run_index –d'{
  "fm_date": "2015-01-09T03:34:17.839651",
   "TotalEvents": 233528,
   "NEvents": 10368,
   "ls": 2,
   "NFiles": 30,
   "id": "run231913_ls0002_EoLS_bu-c2f16-19-01"
}'
```

mapping(document description):

```
"runindex_index" : {
    "mappings" : {
      "eols" : {
        "_id" : { "path" : "id" },
        "_parent" : { "type" : "run"},
        "_timestamp" : {"path" : "fm_date"}.
        "properties" : {
          "NEvents" : {"type" : "integer"},
          "NFiles" : {"type" : "integer"},
          "NLostEvents" : {"type" : "integer"},
          "TotalEvents" : {"type" : "integer"},
          "fm_date" : {"type" : "date",
            "format" : "dateOptionalTime"}},
          "id" : {"ype" : "string","index":"not_analyzed"},
          "ls" : {"type" : "integer"}
        },…
```

query:

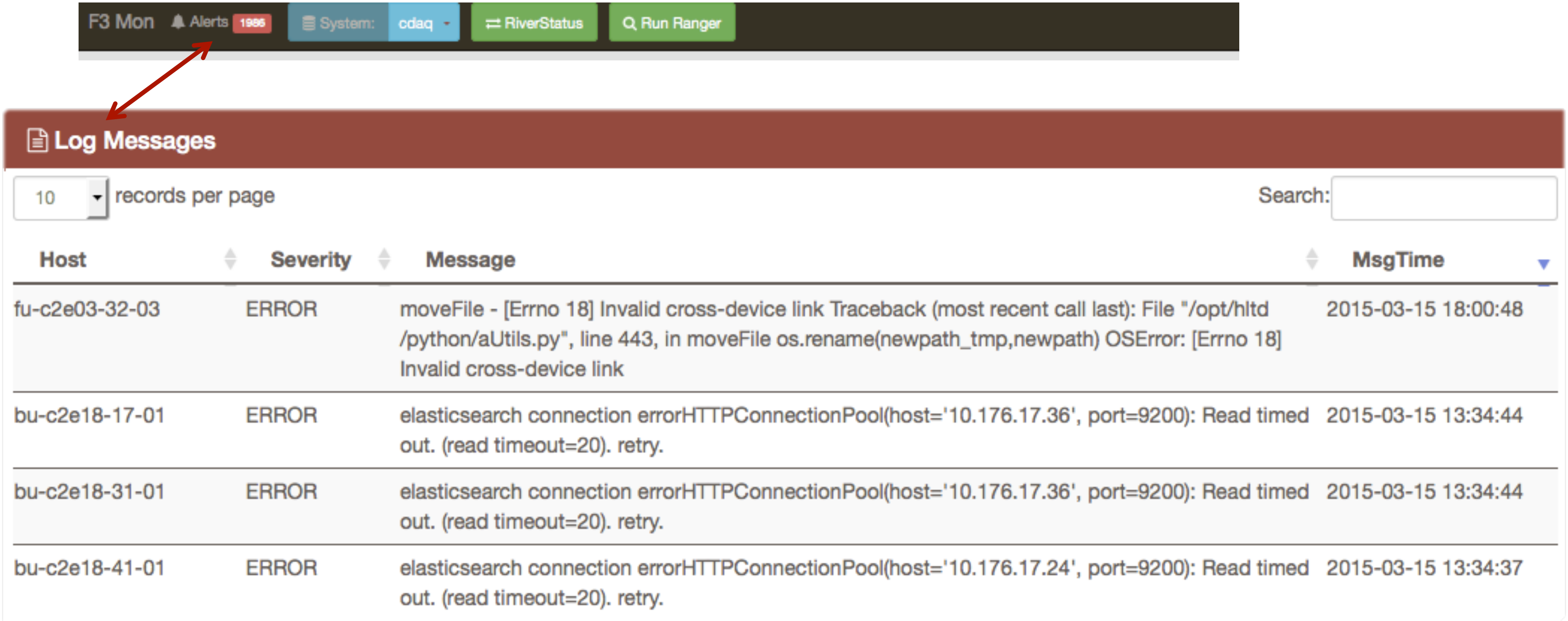
```
curl –XGET http://es-srv:9200/run_index/eols/_search
–d'{
  "query":{
    "term":{"ls":2}
  }
}'
```

result:

```
{    "_index" : "run_index",
     "_type" : "eols",
     "_id" : "run231913_ls0002_EoLS_bu-c2f16-19-01",
     "_score" : 1.0,
     "_source":{
       "fm_date": "2015-01-09T03:34:17.839651",
       "TotalEvents": 233528,
       "NEvents": 10368,
       "ls": 2,
       "NFiles": 30,
       "id": "run231913_ls0002_EoLS_bu-c2f16-19-01"
     }
}
```

Logging

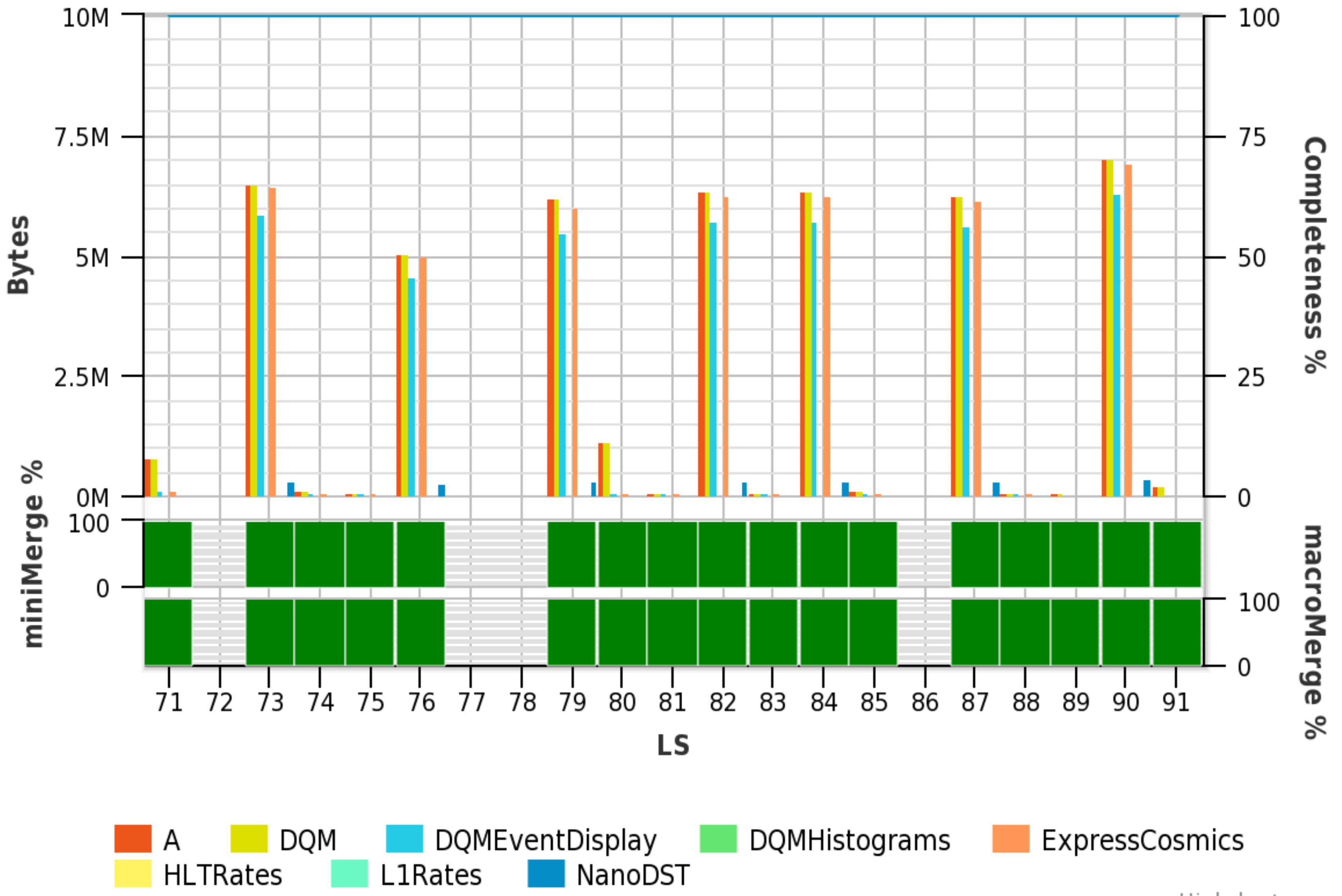
- Logs of FU services (excluding HLT jobs) stored in central-ES cluster
- Documents built by locally running script parsing log files and injected directly into central ES
- UI: logs fetched using a query selecting timestamps within run window



Host	Severity	Message	MsgTime
fu-c2e03-32-03	ERROR	moveFile - [Errno 18] Invalid cross-device link Traceback (most recent call last): File "/opt/hltd /python/aUtils.py", line 443, in moveFile os.rename(newpath_tmp,newpath) OSError: [Errno 18] Invalid cross-device link	2015-03-15 18:00:48
bu-c2e18-17-01	ERROR	elasticsearch connection errorHTTPConnectionPool(host='10.176.17.36', port=9200): Read timed out. (read timeout=20). retry.	2015-03-15 13:34:44
bu-c2e18-31-01	ERROR	elasticsearch connection errorHTTPConnectionPool(host='10.176.17.36', port=9200): Read timed out. (read timeout=20). retry.	2015-03-15 13:34:44
bu-c2e18-41-01	ERROR	elasticsearch connection errorHTTPConnectionPool(host='10.176.17.24', port=9200): Read timed out. (read timeout=20). retry.	2015-03-15 13:34:37

- HLT (CMSSW) logs saved in appliance ES indices

Easter beam splashes in F^{3}Mon


10M
7.5M
5M
2.5M
0M
Bytes
Completeness %
miniMerge %
macroMerge %
LS
A
DQM
DQMEventDisplay
DQMHistograms
ExpressCosmics
HLTRates
L1Rates
NanoDST
Highcharts.com