

Landscape Data

What is there, how do I access it, and what has been shared with Cambridge Analytica?

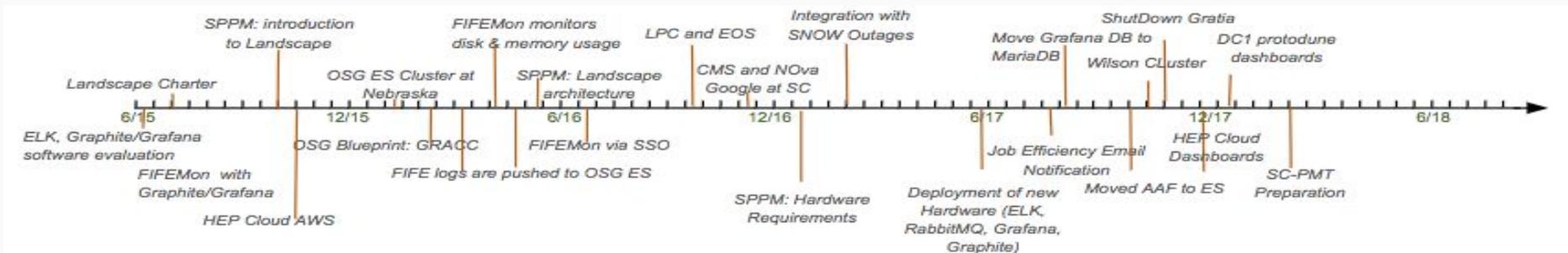
April 19, 2018

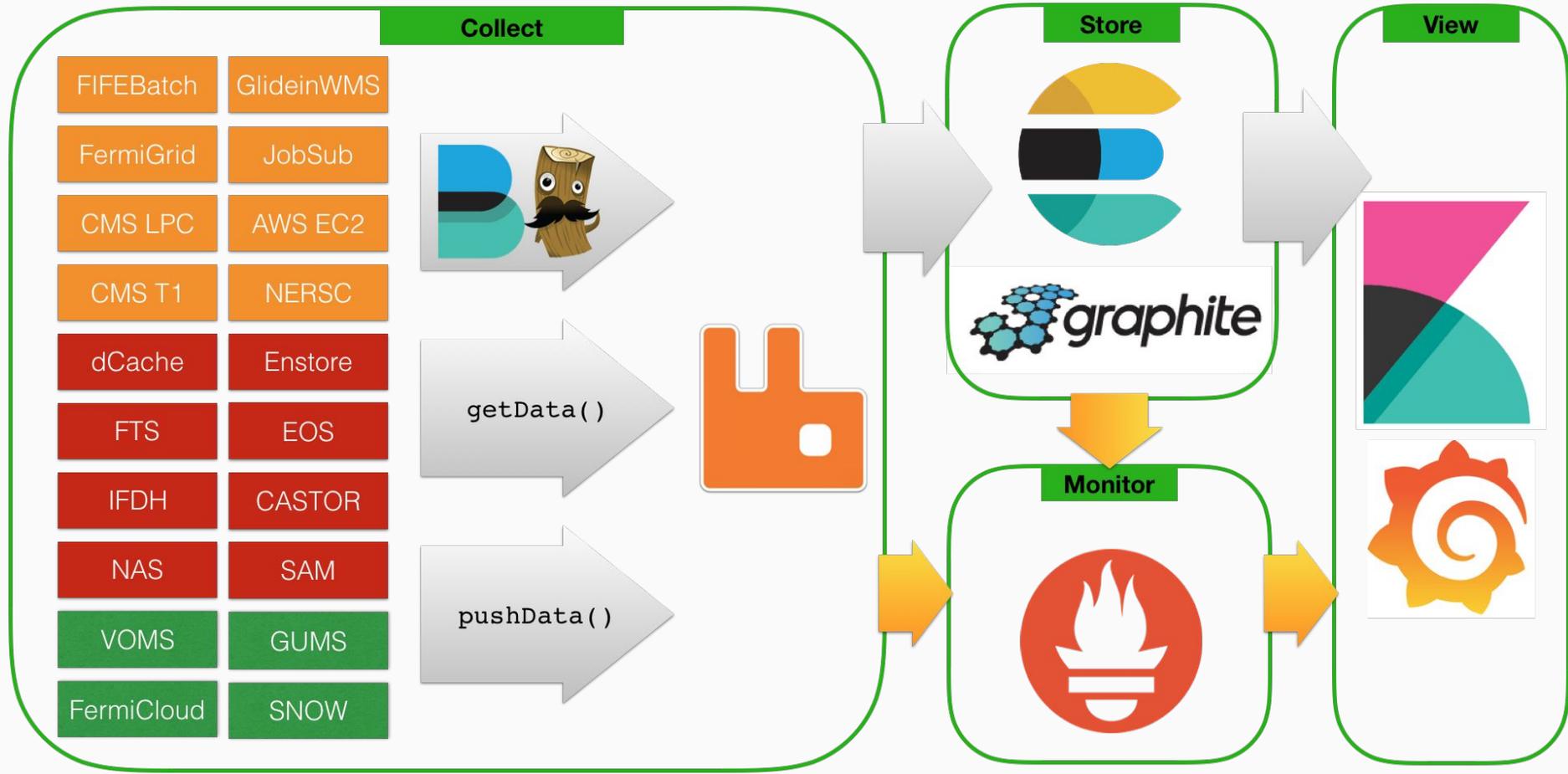


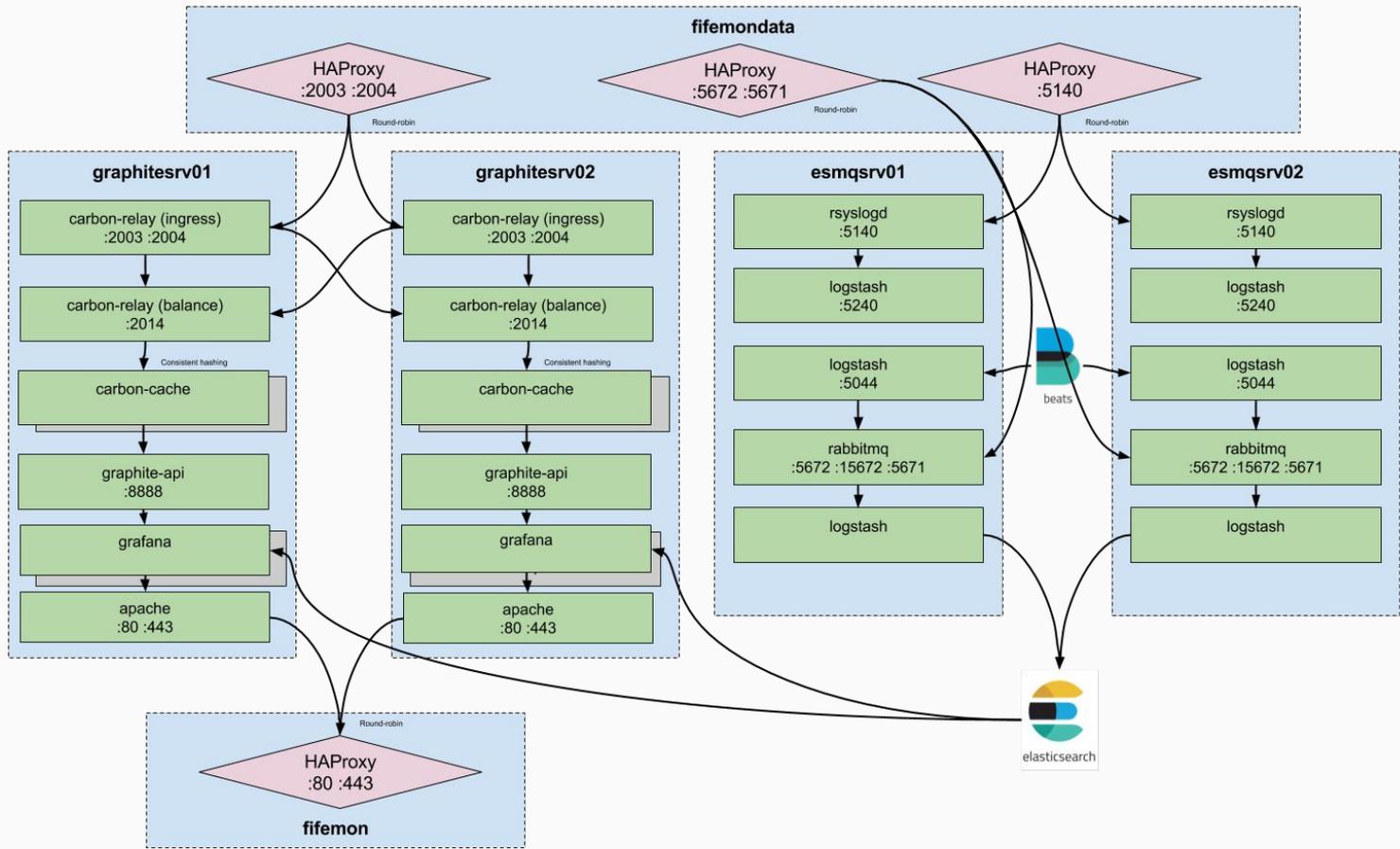
Landscape

Landscape is monitoring program for FIFE, HEPCloud, and more

- FIFE, FermiGrid, dCache: <https://fifemon.fnal.gov/monitor>
- CMS Tier 1 & LPC <https://landscape.fnal.gov/lpc>
- HEPCloud <https://fifemon.fnal.gov/hcf>
- HPC (Wilson Cluster) <https://landscape.fnal.gov/hpc>
- OSG Accounting (limited support) <https://gracc.opensciencegrid.org>







Graphite vs Elasticsearch

Graphite stores time-series metrics:

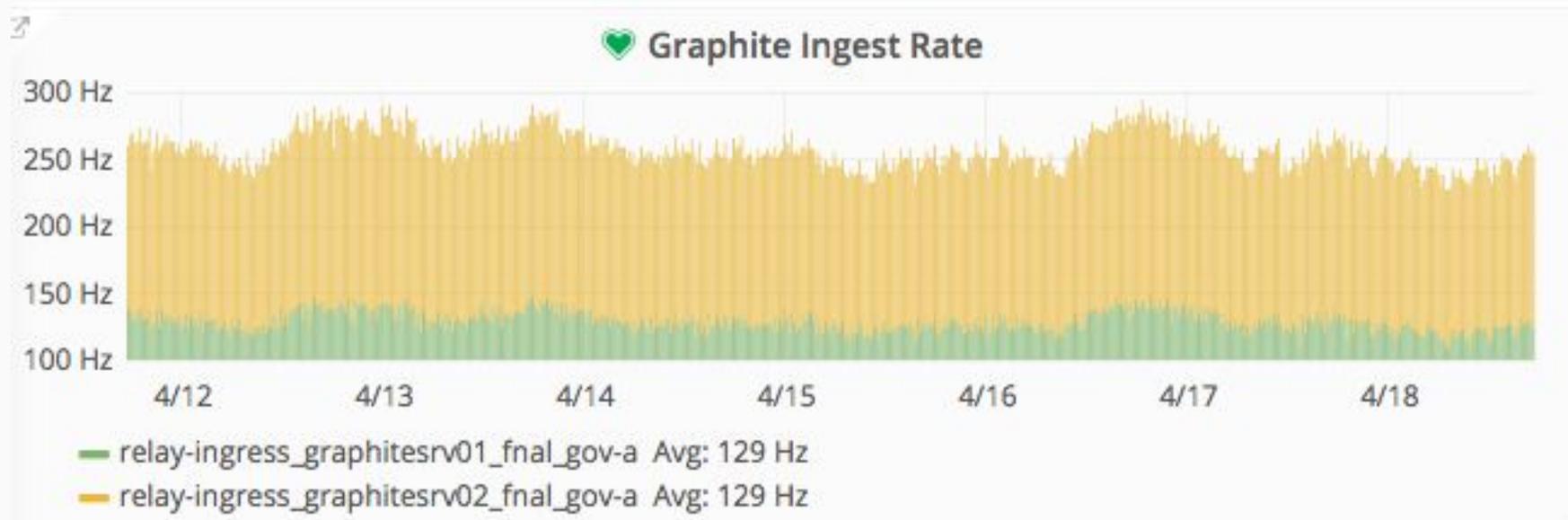
- System state values (e.g. how many jobs are running)
- System performance statistics (e.g. memory usage)

Primary use: short- to long-term trending and alarming on predetermined & constrained dimensions

Elasticsearch stores detailed documents and logs:

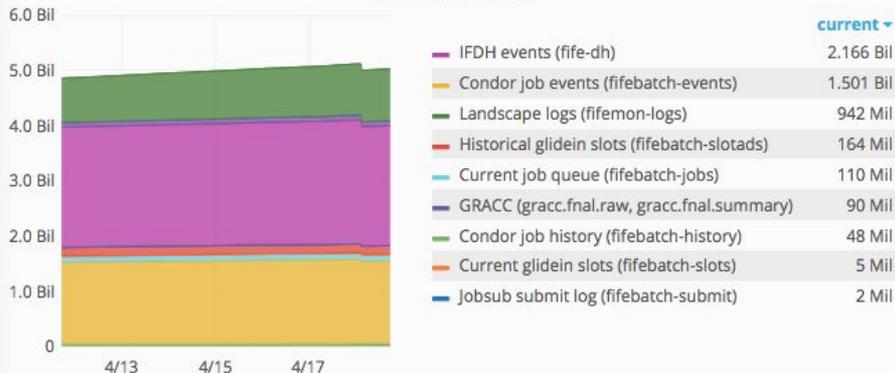
- Current and historical job info
- Current and historical slot (machine) info
- HTCondor events (job state changes)
- IFDH transfer events
- dCache and Enstore transfers (GRACC)
- Landscape service logs

Primary use: short-term trending and analytics on arbitrary dimensions

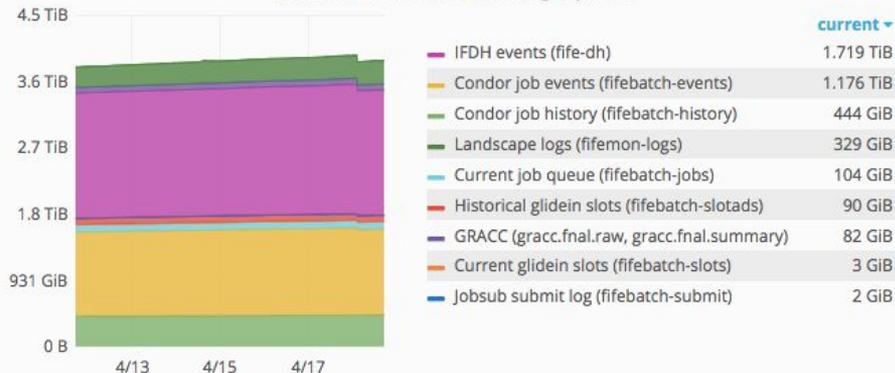


~2.3 million Metrics, ~800 GB on disk

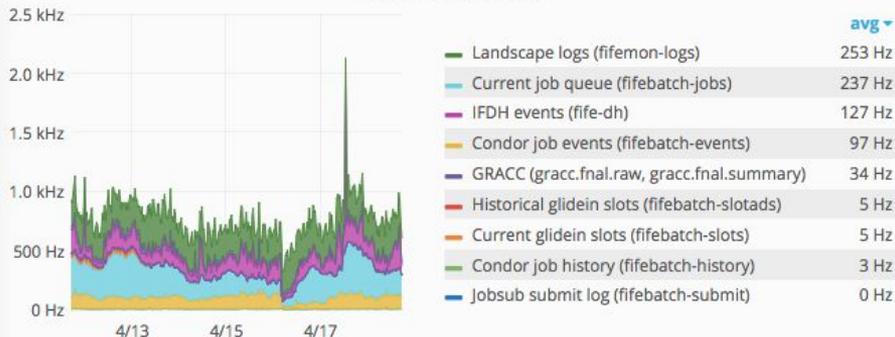
Total Documents



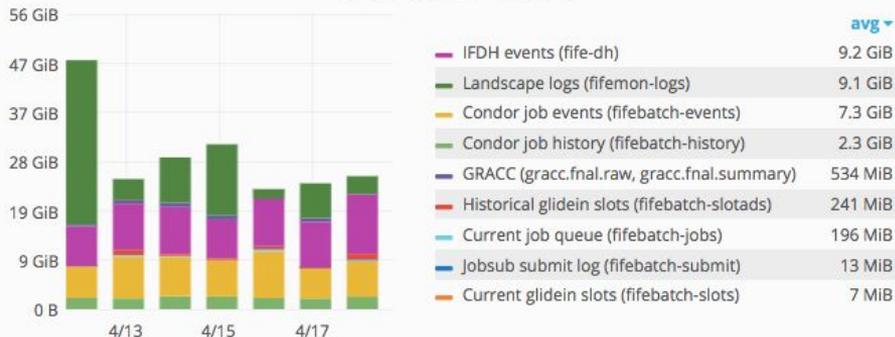
Total Store Size (not including replicas)



Document Index Rate



Daily change in store size



Elasticsearch

Elasticsearch Concepts

- Data model and interchange format is JSON documents composed of multiple key:value fields.
- Data storage engine is Apache Lucene.
- Data is stored in indices, distributed between one or more shards - each shard is a Lucene database.
- Commonly data is stored in time-based index patterns for partitioning and scalability (e.g. fifebatch-logs-YYYY.MM.DD). Clients usually support wildcards (e.g. fifebatch-logs-*), some “smarter” than others.

Elasticsearch Concepts

- An index can have multiple document types (deprecated!)
- A type has defined field mappings that are set at index time. The mapping defines the type of field (string, float, etc) and how it is analyzed. Changing a field mapping requires re-indexing the data!
- By default string fields are analyzed by lowercasing and splitting on word boundaries (space, dash, etc). Great for ordinary text search, not great for identifying strings (e.g. host names).
- Most strings in FIFE indices are “keyword” type that is not analyzed.

Elasticsearch Data

Key indices and fields

fifebatch-jobs: current job data

fifebatch-history: historical job data

fifebatch-events: Job state changes

fifebatch-slots: current glidein data

fife-dh: IFDH transfer events

gracc: dCache, enstore, and HEPCloud accounting

fifebatch-jobs

Select job attributes (ClassAds) updated every 10 minutes from fifebatch HTCondor queue.

- Single alias pointing to one or more indices.
- @timestamp is set to the last time the job was updated.
- Once the job falls out of the queue (complete, cancel) it stops being updated, so a job will remain forever in “running” state. Query “Last 10m” for current jobs.
- Future plans: collect all classads and better integrate with “fifebatch-history” indices

Key Attributes

jobid "123456789.0@fifebatch2.fnal.gov"

cluster "123456789"

process "0"

schedd "jobsub01.fnal.gov"

status

1: idle

2: running

~~3: cancelled~~

~~4: completed~~

5: held

Owner "novapro"

JobsubClientKerberosPrincipal

"amoren@FNAL.GOV"

JobSub_Group "nova"

Request Attributes

DESIRED_usage_model "DEDICATED"

JOB_EXPECTED_MAX_LIFETIME "28800" (s)

RequestCpus "1"

RequestMemory "2048" (MB)

memory_request "2GB" (scripted bytes)

RequestDisk "35000000" (KB)

Cmd "/fife/local/scratch/...2707850_0_1_wrap.sh"

Running Attributes

MATCH_GLIDEIN_Site "FNAL"

walltime "3065" (s)

cputime "2396" (s)

efficiency "0.782"

ResidentSetSize_RAW "1,343,928" (KB)

memory_usage "1.282GB" (scripted bytes)

DiskUsage_RAW "110" (KB)

fifebatch-history

Complete job attributes (ClassAds) representing final state of jobs.

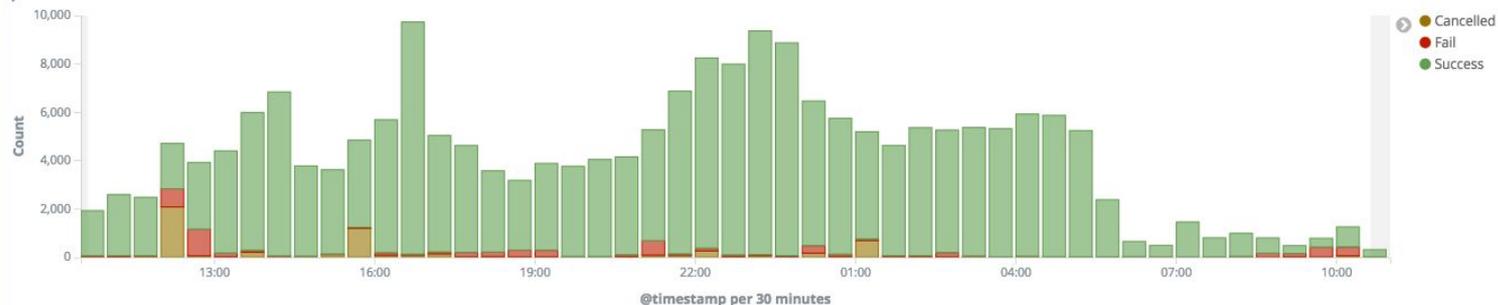
- Daily indices (e.g. fifebatch-history-2018.04.19)
- Keep two years
- Complete “raw” HTCondor ClassAds
- Future plans: add derived fields and transfer information

Search... (e.g. status:200 AND extension:PHP)

Uses lucene query syntax

Add a filter +

Job Error Rate



fifebatch-history Sites

fifebatch-history Groups



fifebatch history jobs

1-50 of 210,346

Time	JobsJobId	Owner	ExitCode	ExitSignal	MATCH_GLIDEIN_Site	MachineAttrMachine0	stdout	stderr
▶ April 19th 2018, 10:37:07.000	6869287.875@jobsub02.fnal.gov	ascarpel	0	-	FermiGrid	fnpc4510.fnal.gov	[stdout]	[stderr]
▶ April 19th 2018, 10:37:07.000	6869287.926@jobsub02.fnal.gov	ascarpel	0	-	FermiGrid	fnpc9040.fnal.gov	[stdout]	[stderr]
▶ April 19th 2018, 10:37:00.000	6869287.891@jobsub02.fnal.gov	ascarpel	0	-	FermiGrid	fnpc5024.fnal.gov	[stdout]	[stderr]

fifebatch-history field reference

Key Attributes

JobsubJobId "123456789.0@fifebatch2.fnal.gov"

ClusterId "123456789"

ProclD "0"

host "jobsub01.fnal.gov"

JobStatus

1: idle

2: running

3: cancelled

4: completed

~~5: held~~

Owner "novapro"

JobsubClientKerberosPrincipal

"amoren@FNAL.GOV"

Jobsub_Group "nova"

Final Attributes

ExitCode/ExitSignal

Success: "ExitCode:0 AND JobStatus:4"

Fail: "NOT ExitCode:0 AND JobStatus:4"

MachineAttrMachine0 "fnpc9060.fnal.gov"

MachineAttrGLIDEIN_Site0 "FermiGrid"

CommittedTime "3065"

RemoteWallClockTime is similar, but includes evictions

RemoteUserCpu "2396" (s)

RemoteSysCpu "12" (s)

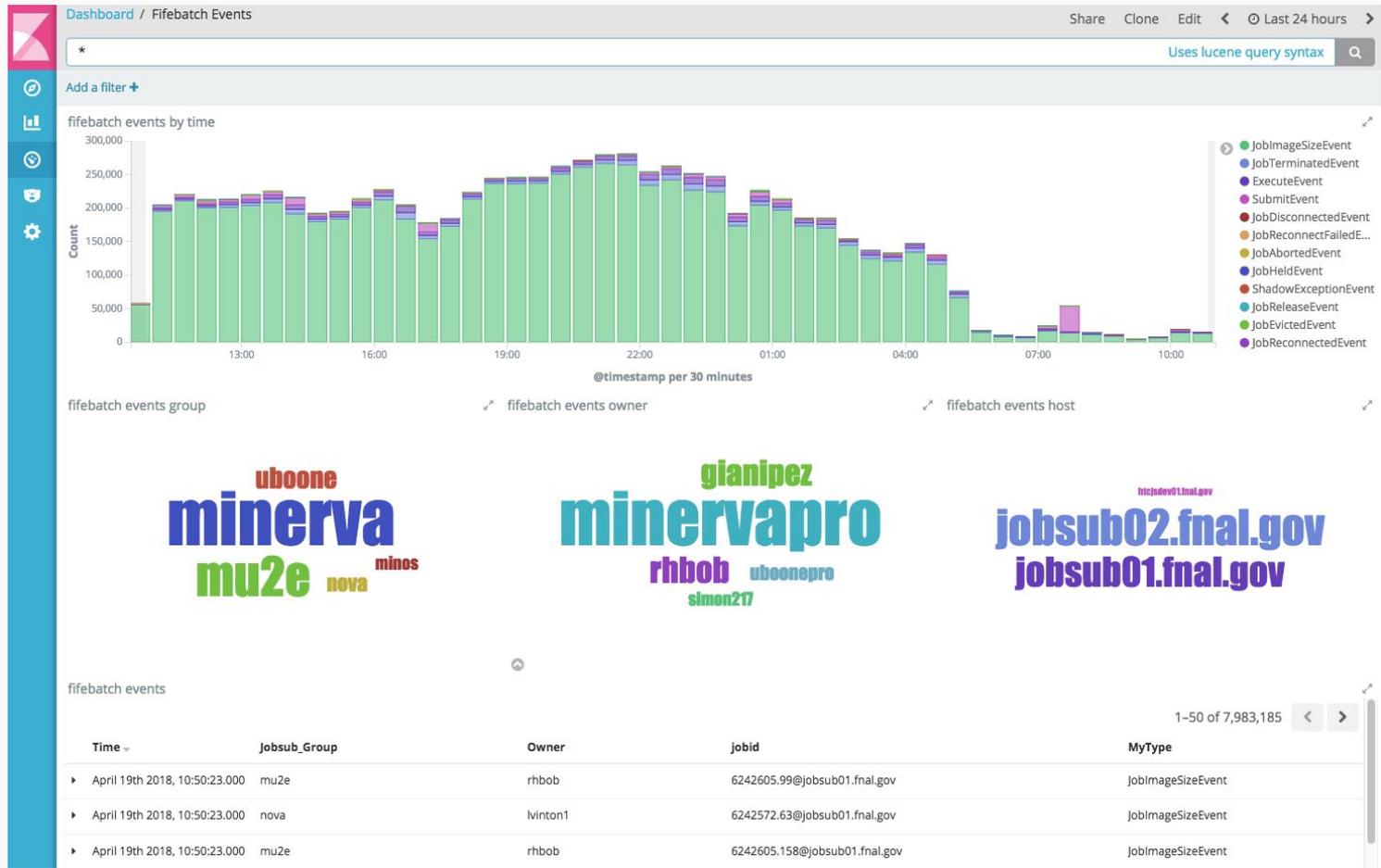
ResidentSetSize_RAW "1,343,928" (KB)

DiskUsage_RAW "110" (KB)

fifebatch-events

HTCondor event logs (job submitted, job started, etc) collected in “real time”

- Daily indices (e.g. fifebatch-events-2018.04.19)
- Keep 180 days



fifebatch-events field reference

Key Attributes

jobid "123456789.0@fifebatch2.fnal.gov"

cluster "123456789"

process "0"

host "fifebatch2.fnal.gov"

Owner "novapro"

JobSub_Group "nova"

MachineAttrGLIDEIN_Site0 "FNAL"

MachineAttrGLIDEIN_ResourceName0 "GPGrid"

MachineAttrMachine0 "fnpc9060.fnal.gov"

Event Attributes

TriggerEventTypeName "5"

TriggerEventName

"ULOG_JOB_TERMINATED"

MyType "JobTerminatedEvent"

Key Events

1	SubmitEvent
2	ExecuteEvent
4	JobEvictedEvent
5	JobTerminatedEvent
6	JobImageSizeEvent
9	JobAbortedEvent
12	JobHeldEvent
13	JobReleasedEvent
22	JobDisconnectedEvent
23	JobReconnectedEvent

fifebatch-slots

Select slot/machine attributes (ClassAds) updated every 10 minutes from fifebatch HTCondor collector.

- Single alias pointing to one or more indices.
- Similar to jobs, once the slot disappears it stops being updated, so a slot will remain forever in “Claimed” state. Query “Last 10m” for current slots.

Key Attributes

Name "slot1_3@glidein_1045375_1086175008@fnpc4201.fnal.gov"

GLIDEIN_Site "FNAL"

GLIDEIN_ResourceName "GPGrid"

SlotType

"Partitionable": pilot

"Dynamic": claimed

State "Claimed"

time_left "295596" (s)

time_left_hours "82"

time_left_days "3"

Resources

For "Partitionable" slot this is what's remaining unclaimed, for "Dynamic" this is what's claimed by the job.

Cpus "35GB"

Disk "35GB"

Memory "3GB" (raw in bytes)

Memory_mb "3072"

Memory_gb "3"

FifemonSlotWeight "1.5"

FifemonSlotWeight is fractional "standard" slots - 1 Core, 2GB RAM

fife-dh

IFDH event logs (start transfer, end transfer, POMS data) collected in “real time”

- Daily indices (e.g. fife-dh-2018.04.19)
- Keep 180 days
- Coming soon: better parse existing events and consistent field names with HTCondor
- Future plans: revamp IFDH logging to include more useful information

Search... (e.g. status:200 AND extension:PHP)

Uses lucene query syntax

Add a filter +

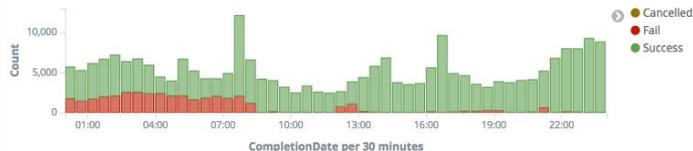
fifebatch-history Groups



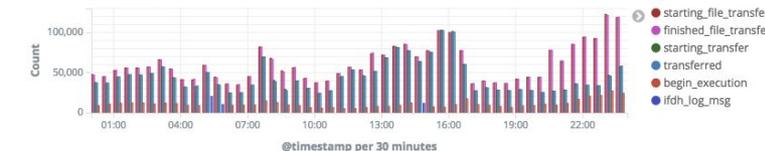
history owner tag cloud



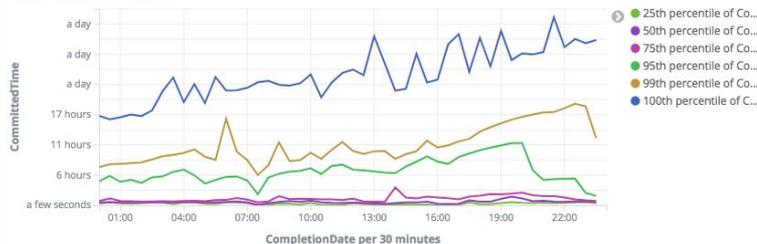
Job Error Rate



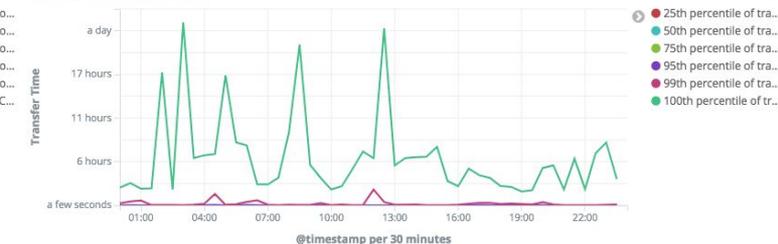
IFDH Transfer Events



history walltime percentiles



ifdh transfer time percentiles



failed jobs per site

MachineAttrGLIDEIN_Site0: Descending	Count
FermiGrid	37,866
Colorado	149
UChicago	50
Wisconsin	27

failed jobs per host

MachineAttrMachine0: Descending	Count
fnpc9044.fnal.gov	224
fnpc9057.fnal.gov	211
fnpc9058.fnal.gov	204
fnpc17135.fnal.gov	198

failed jobs per owner

Owner: Descending	Count
rhbob	31,528
gianipez	2,000
cbsykes	1,797
minervapro	723

fife-dh field reference

Key Attributes

jobid/JobsubJobId

“123456789.0@fifebatch2.fnal.gov”

cluster “123456789”

process “0”

user/Owner “novapro”

experiment/Jobsub_Group “nova”

ifdh_event_type “starting_transfer”

Transfer Attributes

node_name “fnpc6013.fnal.gov”

sourcepath

“/pnfs/GM2/scratch/.../gm2offline_reco_15395.00068.root ”

destpath

“/pnfs/GM2/scratch/.../gm2offline_reco_15395.00068.root ”

bytes_transferred “2,016,223”

transfer_time “4.78027” (s)

Event Types

starting_transfer	“Ifdh cp” start
transferred	“Ifdh cp” finished
starting_file_transfer	Start of single file transfer
finished_file_transfer	End of single file transfer
begin_execution	Job started
finished	Job finished
failed_transfer	Failed transfer
poms_data	POMS metadata for job
ifdh_log_msg	Catch-all

gracc.fnal

Grid accounting system for OSG: <https://gracc.opensciencegrid.org>

Landscape maintains FNAL internal accounting of:

- dCache and enstore transfers
- Wilson cluster
- Fermicloud
- HEPCloud

Documentation: <https://opensciencegrid.github.io/gracc/>

Accessing Data

Kibana

cURL

Client libraries

Landscape API?

Kibana

Elasticsearch's native data browser and analytics platform.

<https://landscapeitb.fnal.gov/kibana> (open with FNAL SSO login)

Or

<https://landscape.fnal.gov/kibana> (currently restricted)

Discover:
Search and
Filter raw
documents

Visualize:
Graph
document
statistics

Dashboard:
Combine
visualizations

The screenshot shows the Kibana interface with the following elements and callouts:

- Discover:** A callout points to the 'Discover' button in the left sidebar.
- Visualize:** A callout points to the 'Visualize' button in the left sidebar.
- Dashboard:** A callout points to the 'Dashboard' button in the left sidebar.
- Search/filter:** A callout points to the search bar and filter controls at the top of the main panel.
- Toolbar:** A callout points to the 'New', 'Save', 'Open', and 'Share' buttons in the top right of the main panel.
- Timepicker:** A callout points to the time range selector (April 18th 2018, 11:04:24.140 - April 19th 2018, 11:04:24.140) and the 'Auto' dropdown.

The main panel displays a search for 'fife-dh-*' with 10,191,763 hits. A bar chart shows the distribution of results over time. Below the chart, a list of fields is shown: sourcepath, tags, transfer_rate, transfer_time, user, and @timestamp. A sample log entry is displayed at the bottom:

```
April 19th 2018, 11:04:21.000 bytes_transferred: 0 process_id: 2789 ifdh_version: v2_1_0 n
ode_name: gm2samgpvm01.fnal.gov ifdh_use: interactive message
: 2018-04-19T16:04:21Z gm2samgpvm01.fnal.gov gm2pro/gm2/v2_1_0/g
m2[2789]: ifdh: ifdh cp: transferred: 0 bytes in 13.1062 seconds
type: ifdh tags: transferred, interactive @timestamp: April 1
```

Kibana tips

- Everything is interactive! Click on a legend value or in a table to filter.
- Filters can be “pinned” to carry between views
- Kibana supports powerful full-text search queries using Elasticsearch [query_string query](#), which are in turn based on Lucene expressions.
 - Specify field to search against as “field:term”
 - Combine terms with boolean operations (OR, AND, NOT - caps are important!)
 - Group terms with parens “name:(foo bar)” (OR is implicit)
 - Match wildcards “name:foo*” and regex “name:/foo.*”
 - Numeric comparisons “Memory_mb:>2000”

curl

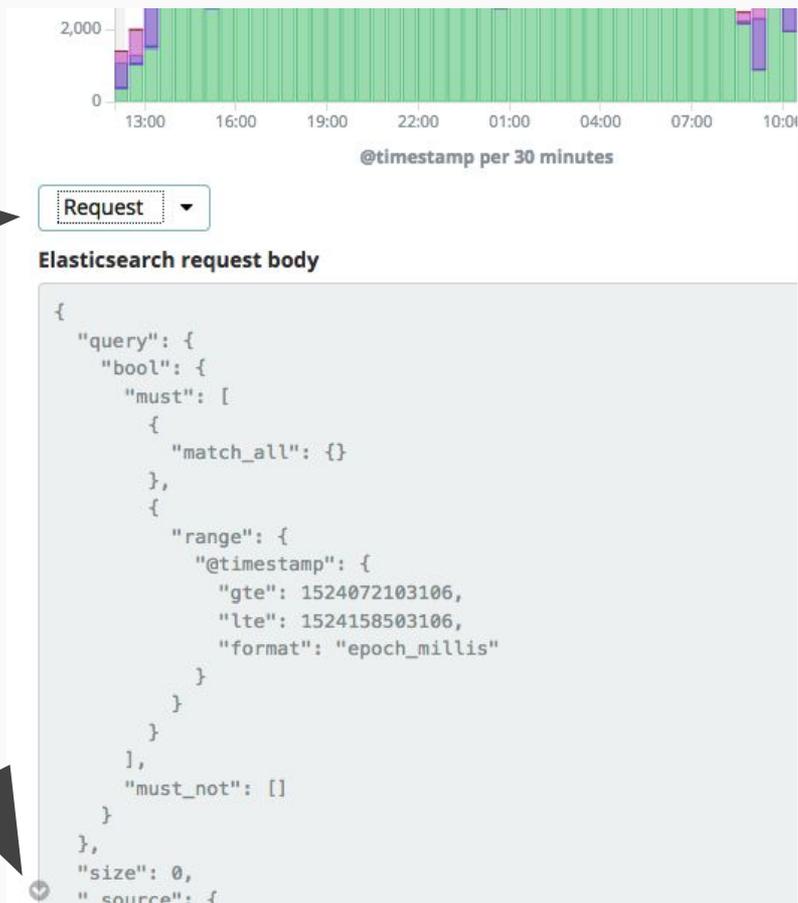
Elasticsearch uses a RESTful HTTP interface, we expose that read-only at <https://fifemon-es.fnal.gov>

The Elasticsearch [query DSL](#) is quite complex JSON-based language, but it's very powerful.

In addition to searching for documents, it can do aggregations on numeric fields, group results into nested buckets, and more.

Protip:

Click the “expand” arrow at the bottom of a Kibana visualization then select “Request” to see the raw query that Kibana is making to Elasticsearch.



Get a single document by ID

```
$ curl 'https://fifemon-es.fnal.gov/fifebatch-jobs/job/6185988.0@jobsub01.fnal.gov?pretty'  
{  
  "_index" : "fifebatch-jobs.01",  
  "_type" : "job",  
  "_id" : "6185988.0@jobsub01.fnal.gov",  
  "_version" : 576,  
  "found" : true,  
  "_source" : {  
    "cluster" : 6185988,  
    "Owner" : "adi",  
    "RequestDisk" : 35000000,  
    "NumJobStarts" : 1,  
    "HoldReasonCode" : 26,  
    "RemoteUserCpu" : 4982.0,  
    "time_ratio" : 0.0,  
    "type" : "job",  
    ...  
  }  
}
```

Search for documents

```
$ curl 'https://fifemon-es.fnal.gov/fifebatch-history-*/_search?pretty -d '
{
  "query": {
    "query_string": {
      "query": "Owner:novapro AND NOT ExitCode:0 AND EnteredCurrentStatus:[now-12h TO now]"
    }
  },
  "size": 1,
  "_source": [
    "JobsubJobId"
  ]
}'
{
  "took" : 96,
  "timed_out" : false,
  "hits" : {
    "total" : 15,
    "hits" : [
      {
        "_index" : "fifebatch-history-2018.04.19",
        "_type" : "condor_history",
        "_id" : "AWLc4W_HXgSn6-mb-4kT",
        "_score" : 6.5376205,
        "_source" : {
          "JobsubJobId" : "6239238.0@jobsub01.fnal.gov"
        }
      }
    ]
  }
}
```

Aggregate

```
$ curl 'https://fifemon-es.fnal.gov/fifebatch-history-*/_search?pretty' -d '{
  "query": {
    "query_string": {
      "query": "Owner:novapro AND NOT ExitCode:0 AND EnteredCurrentStatus:[now-12h TO now]"
    }
  },
  "size": 0,
  "aggs": {
    "walltime": {
      "sum": {
        "field": "CommittedTime"
      }
    }
  }
}'
{
  "took" : 138,
  "timed_out" : false,,
  "hits" : {
    "total" : 15,
    "max_score" : 0.0,
    "hits" : [ ]
  },
  "aggregations" : {
    "walltime" : {
      "value" : 353307.0
    }
  }
}
```

Client Libraries

There are many client libraries to help communicate with and query Elasticsearch.

For example, [elasticsearch](#) and [elasticsearch-dsl](#) for Python.

Example:

<https://opensciencegrid.github.io/gracc/user/direct/#python-elasticsearch-dsl>

Landscape API?

Up to now Elasticsearch and Graphite have really been “internal” services for Landscape.

We don't offer an SLA and we often make breaking changes in index names, field names, metrics namespaces, etc.

If there's interest in using this data outside our monitoring, we probably need to create a stable API. So how do you want to use this data? Let us know!