OAK RIDGE National Laboratory

Advancing ODA standardization through an open source dashboard

Tim Osborne, Rachel Palumbo, Leah Huk, Ryan Adamson, Rob Jones, Corwin Lester

National Center for Computational Sciences (NCCS)

Oak Ridge National Laboratory (ORNL)

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

Introduction

Operational Data Analytics (ODA)

ODA is the continuous

monitoring,
archiving, and
analysis of

near **real-time performance data**, providing actionable information for multiple operational uses.

Relevant Data

To answer questions about our systems, we must be able to select the **relevant data**:

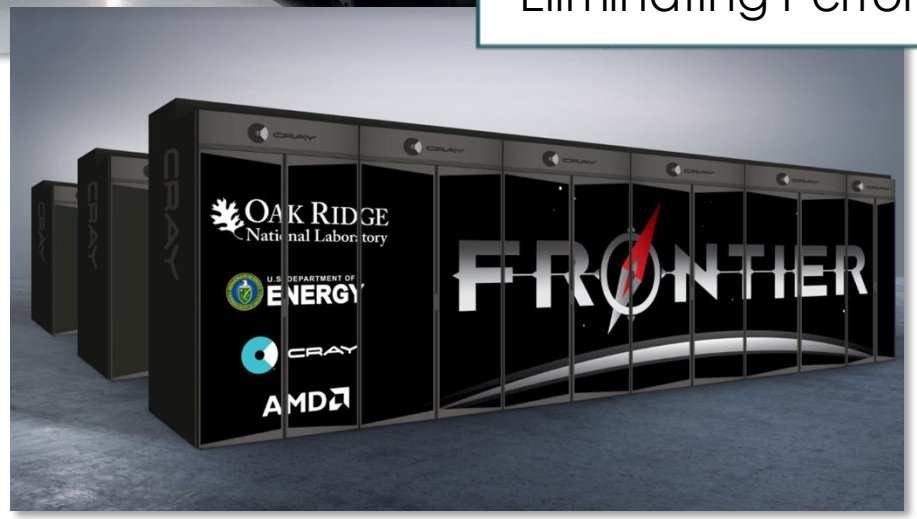
- **Identify** important pieces of information
- **Transform** the data
- **Normalize** vendor specific naming conventions and units of measure
- **Monitor** data quality to prevent missing and incorrect data

Dashboards

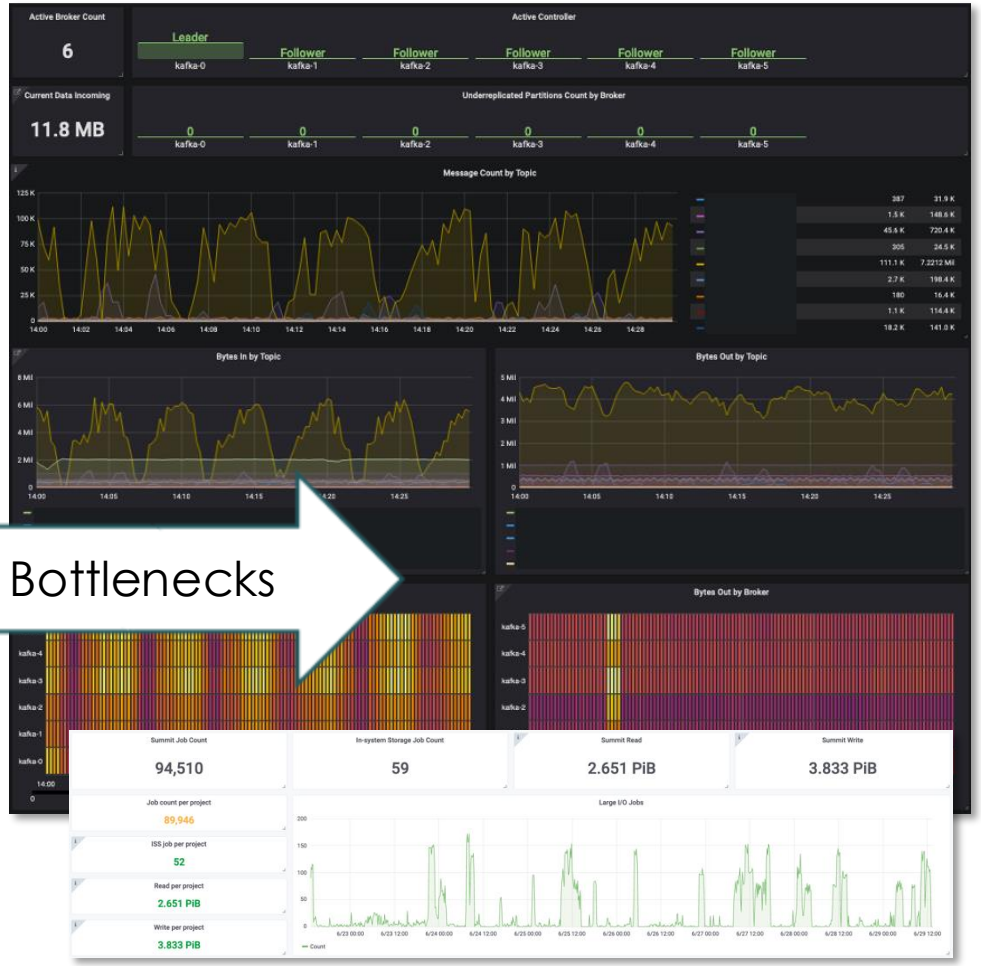
Provide diagnostic and descriptive information across the ODA pillars of:

- Building infrastructure
- System hardware
- System software
- Applications

Hardware and Application Monitoring



Eliminating Performance Bottlenecks



OAK RIDGE National Laboratory

Motivation

Difficulty adapting reporting methods from earlier systems

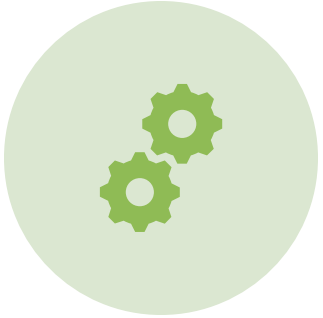
- Operations personnel and users' priorities revolve around:
  - ***Temperature management***
  - ***Power attribution*** *at the job level*
  - ***Node and job level event information*** *synthesis*
- Differences in:
  - **Data format**
  - **Density**
  - **Component granularity**

Goals

*Establish a **standard dashboard** for telemetry data for use at multiple HPC Sites*

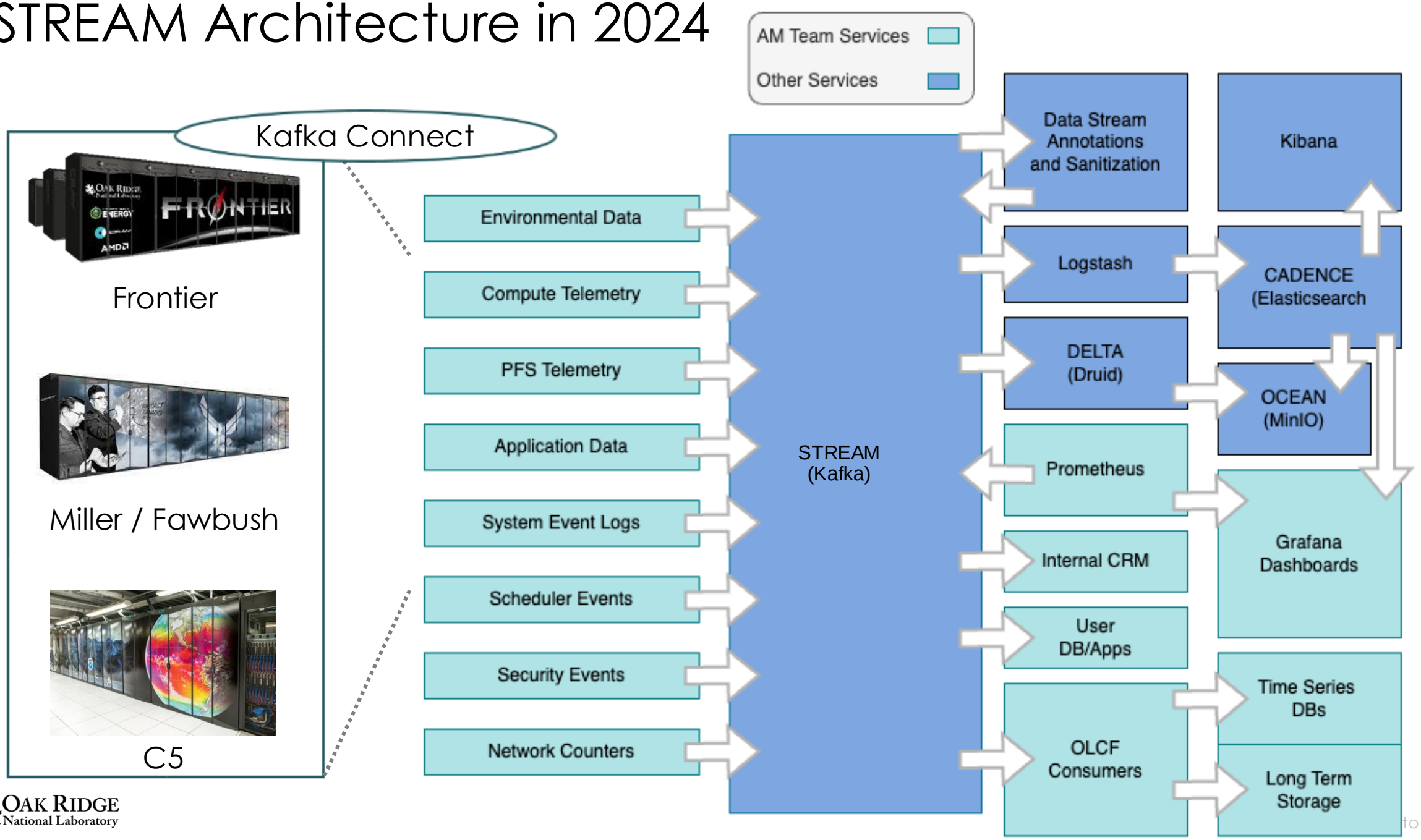
***Reduce the effort** required for HPC sites to understand and use this data effectively*

***Open source** for HPC sites and the ODA community to use and address shared challenges.*



*Provide as a **resource** for vendors as they develop telemetry solutions*

STREAM Architecture in 2024



Input data

SLURM

Custom Slurm scrapers collect **job metrics**

HPE **HPCM**

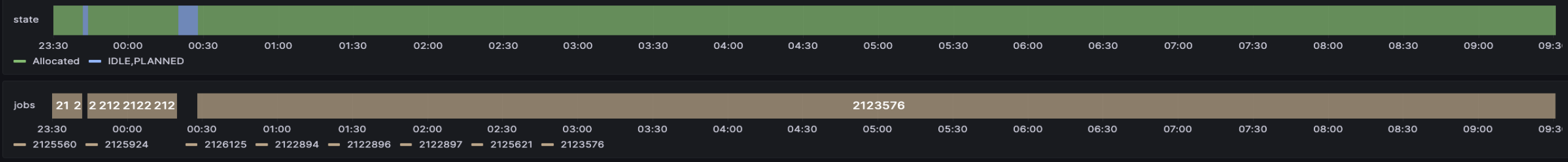
Kafka Connect produces **telemetry data**

Node Info (frontier00001)

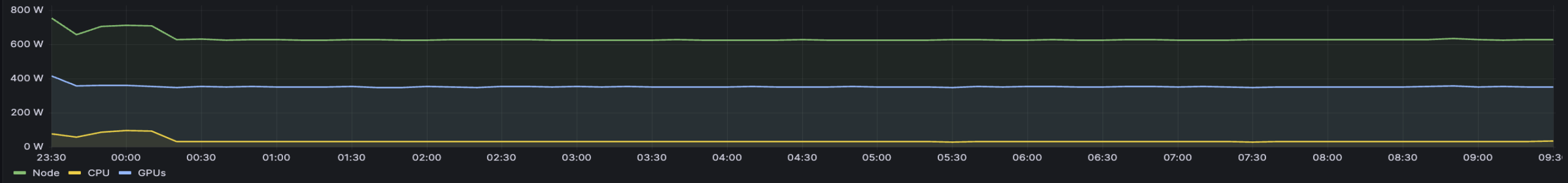
Name	XNAME	Boot Time	CPU Load	Current Power Usage	GCD Temp. - Loop 1	GCD Temp. - Loop 2
frontier00001	x2000c0s0b0n0	Jul 23 2024 18:29:51	1.02%	620 W	6: 39 °C; 4: 40 °C; 7: 43 °C; 5: 48 °C	2: 40 °C; 0: 42 °C; 3: 39 °C; 1: 48 °C

Flags	JobID	Partitions	Used RAM	Current CPU Temp
ALLOCATED	2123576	batch,batch-spi,testing	13.2 GB	41.1 °C

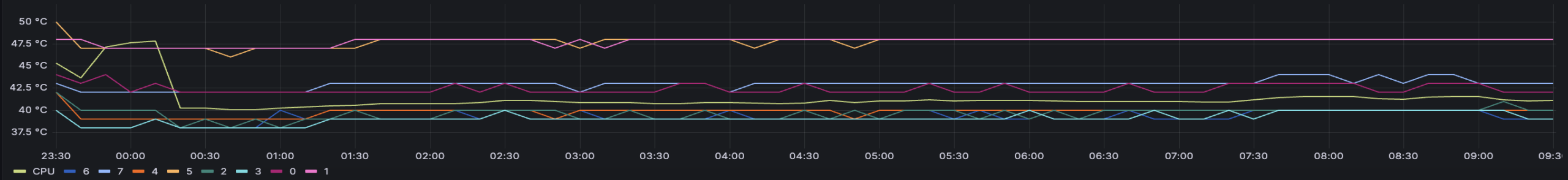
Node Status



Node Power Consumption



Device Temps for frontier00001



Job Info (2123576)

Job ID

2123576

Submit Datetime

07/23/2024, 11:21:22 AM

Login Host

login04

Current State

RUNNING

Batch Host

frontier00001

User

user_name

Project

abc123

Run Time

08:58:53

Node Count

2048

Job Name

job_name

Requested Wall Time

10:00:00

Cores / Node

112

Queue

batch

Job Type

batch job

Exit Code

0:0

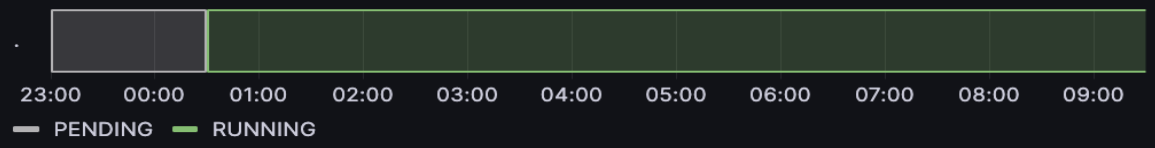
Queue Time

13:06:39

Queue Order

n/a

State Progress



Priority

43679044

State Reason

n/a

Start Time

2024-07-24 00:29:08

End Time

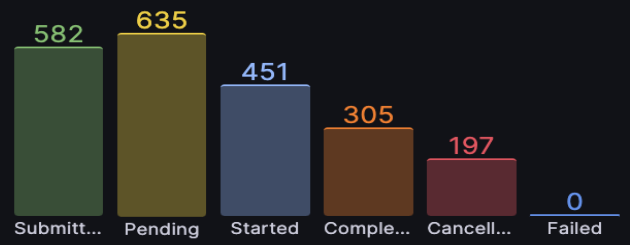
2024-07-24 10:29:08

Der. Exit Code

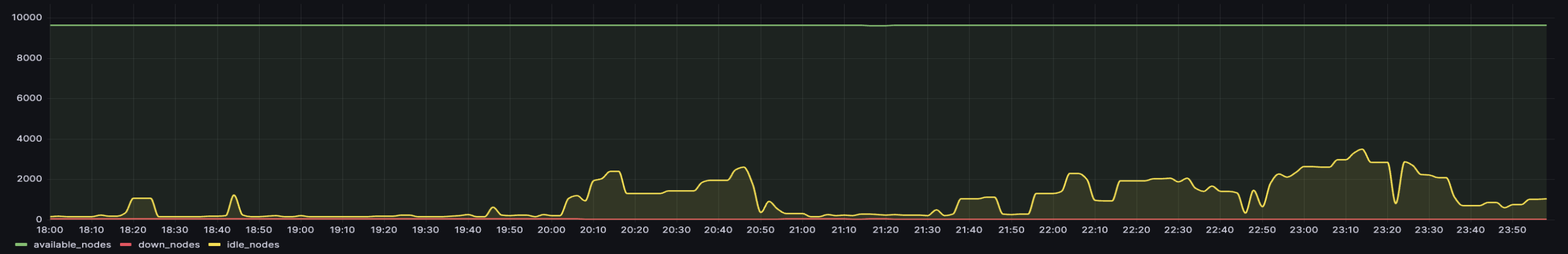
0:0

Live System Statistics

Total Nodes	Available Nodes	Allocated Nodes	Down Nodes	Running Jobs
9408	9372	9276	36	119

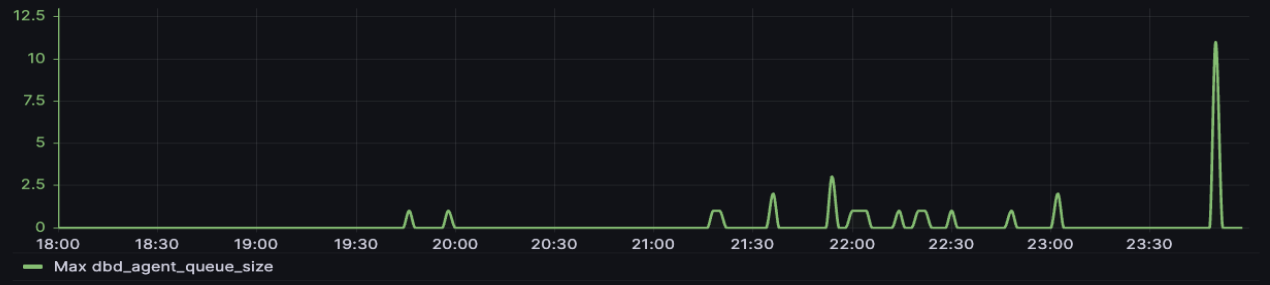


Allocated Nodes History

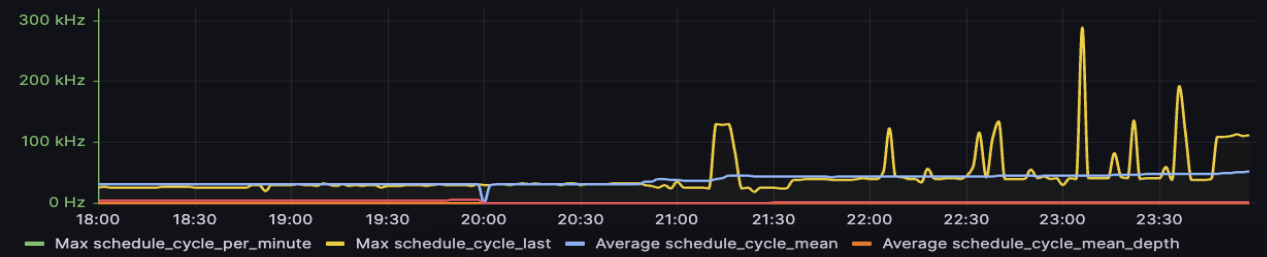


Scheduler Thread Count

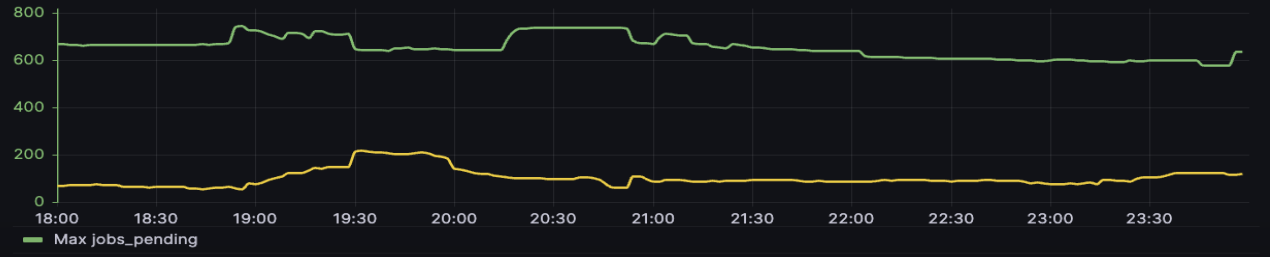
DBD Queue Size



Scheduler Cycles

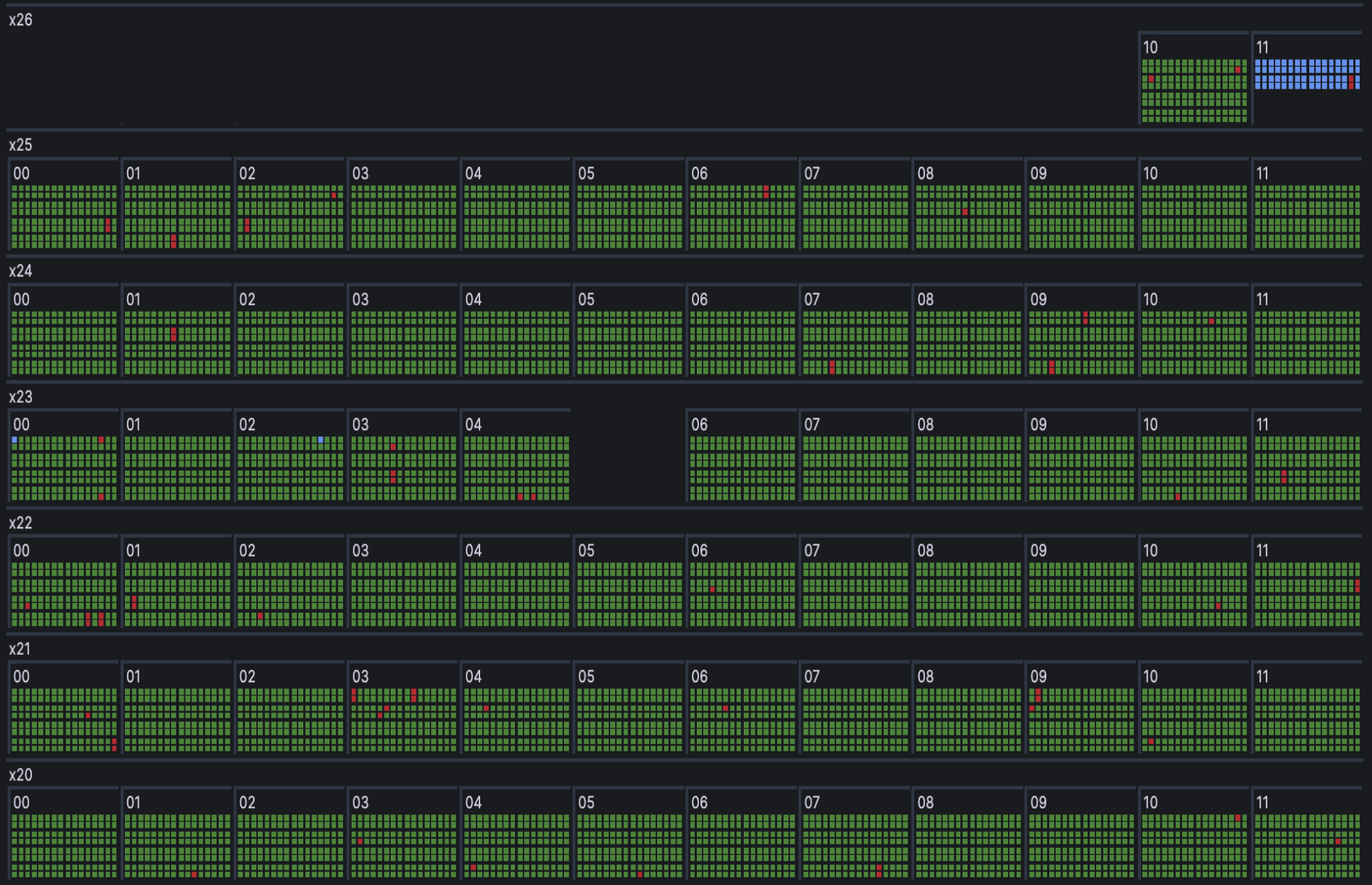


Job Stats



Live System Maps & Queues

Frontier Node Availability



Queued Jobs

Queue Time	Priority ↓	Job ID	Nodes	Wall Time
22:34:47	44144973	2123314	8000	02:00:00
3 d 09:29:46	43891200	2113482	7500	00:20:00
04:03:25	43732931	2125621	4096	02:00:00
01:36:21	43724179	2125924	4096	00:15:00
01:19:04	43723059	2125964	4096	00:15:00
00:23:32	43719725	2126125	2048	00:15:00
13:07:26	43679044	2123576	2048	10:00:00
00:47:52	43547982	2126126	200	02:00:00
00:13:31	43546172	2126550	372	02:00:00
00:06:43	43545733	2126421	64	02:00:00
00:01:34	43545600	2126395	300	01:15:00
00:02:02	43545600	2126428	216	02:00:00
00:00:31 ago	43545600	2126445	6	02:00:00
00:00:36 ago	43545600	2126446	4	00:04:00
00:01:00	43545600	2126555	16	02:00:00
00:00:38	43545600	2126609	2	02:00:00
00:00:47	43545600	2126610	2	01:00:00
00:10:44	43545600	2126616	120	02:00:00
00:02:51	43545600	2126652	16	00:10:00
00:00:37	43545600	2126658	16	00:10:00

Count: 1519

Job Layout (2123576)

Node List

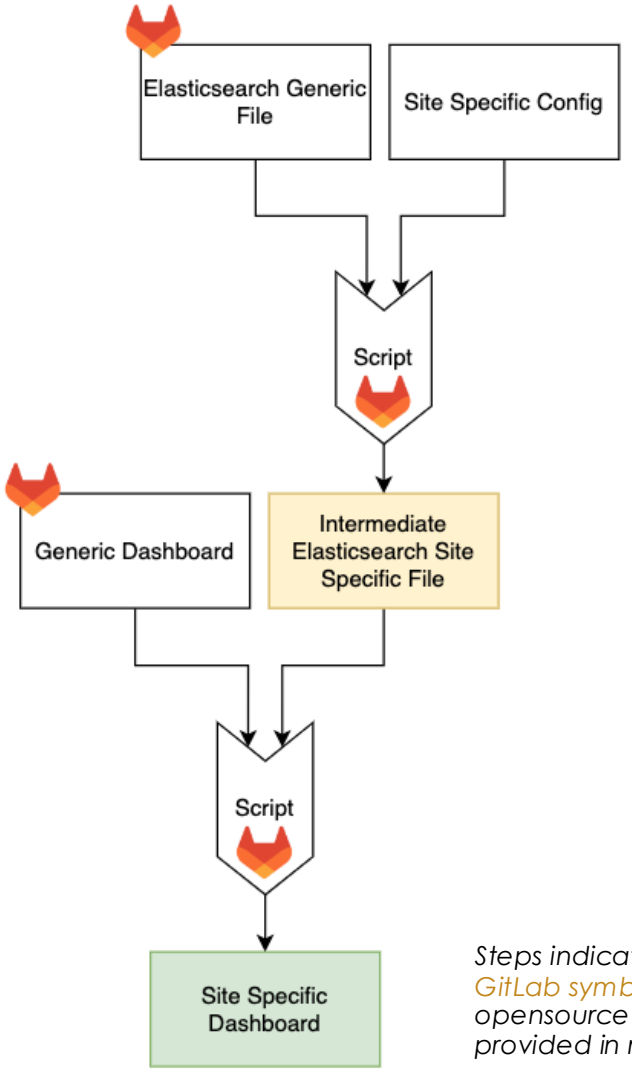
Hostname

frontier00001
frontier00002
frontier00003
frontier00004
frontier00005
frontier00006
frontier00007
frontier00008
frontier00013
frontier00031
frontier00034
frontier00037
frontier00040
frontier00041
frontier00043
frontier00049
frontier00052
frontier00053
frontier00054

Count: 2048

Workflow

- Workflow for building a **site-specific dashboard.**
- Create and maintain your own site specific config
- Generate Elasticsearch Site Specific File
- Create Site Specific Dashboard JSON
- Additional wrapper script is underdevelopment that will generate a new dashboard or update an existing one with the site specific dashboard json



Steps indicated with the GitLab symbol indicate opensource files or scripts provided in repository.

Site Specific Dashboard JSON Model

```
"templating": {
 "list": [
  {
    "current": {
      "selected": false,
      "text": "frontier.node-info",
      "value": "frontier.node-info"
    },
    "hide": 2,
    "includeAll": false,
    "multi": false,
    "name": "node_info",
    "options": [],
    "query": "elasticsearch",
    "refresh": 1,
    "regex": "/^frontier\\.node-info$/",
    "skipUrlSync": false,
    "type": "datasource"
  }
 ]
}
```

Site Specific Dashboard JSON Model

```
"templating": {
 "list": [
  {
   "current": {
    "selected": false,
    "text": "frontier.node-info",
    "value": "frontier.node-info"
   },
   "hide": 2,
   "includeAll": false,
   "multi": false,
   "name": "node_info",
   "options": [],
   "query": "elasticsearch",
   "refresh": 1,
   "regex": "/^frontier\\.node-info$/",
   "skipUrlSync": false,
   "type": "datasource"
  }
 ]
}
```

Site Specific Dashboard JSON Model

```
"templating": {
 "list": [
  {
   "current": {
    "selected": false,
    "text": "frontier.node-info",
    "value": "frontier.node-info"
   },
   "hide": 2,
   "includeAll": false,
   "multi": false,
   "name": "node_info",
   "options": [],
   "query": "elasticsearch",
   "refresh": 1,
   "regex": "/^frontier\\.node-info$/",
   "skipUrlSync": false,
   "type": "datasource"
  }
 ]
}
```

Elasticsearch Generic File

```
"templating": {
  "list": {
    "ID_KEY": "name",
    "node_info": {
      "query": "elasticsearch",
    },
  }
}
```

Site Specific Dashboard JSON Model

```
"templating": {
 "list": [
  {
    "current": {
      "selected": false,
      "text": "frontier.node-info",
      "value": "frontier.node-info"
    },
    "hide": 2,
    "includeAll": false,
    "multi": false,
    "name": "node_info",
    "options": [],
    "query": "elasticsearch",
    "refresh": 1,
    "regex": "/^frontier\\.node-info$/",
    "skipUrlSync": false,
    "type": "datasource"
  }
 ]
}
```

Site Specific Config File

```
"templating": {
  "list": {
     "node_info":{
        "current":{
          "text":"frontier.node-info",
          "value":"frontier.node-info"
        },
        "regex": "/^frontier\\.node-info$/"
     }
  }
}
```

Intermediate Elasticsearch Site Specific File

```
"templating": {
  "list": {
    "ID_KEY": "name",
    "node_info": {
      "query": "elasticsearch",
      "current":{
        "text":"frontier.node-info",
        "value":"frontier.node-info"
      },
      "regex": "/^frontier\\.node-info$/"
    },
  }
}
```

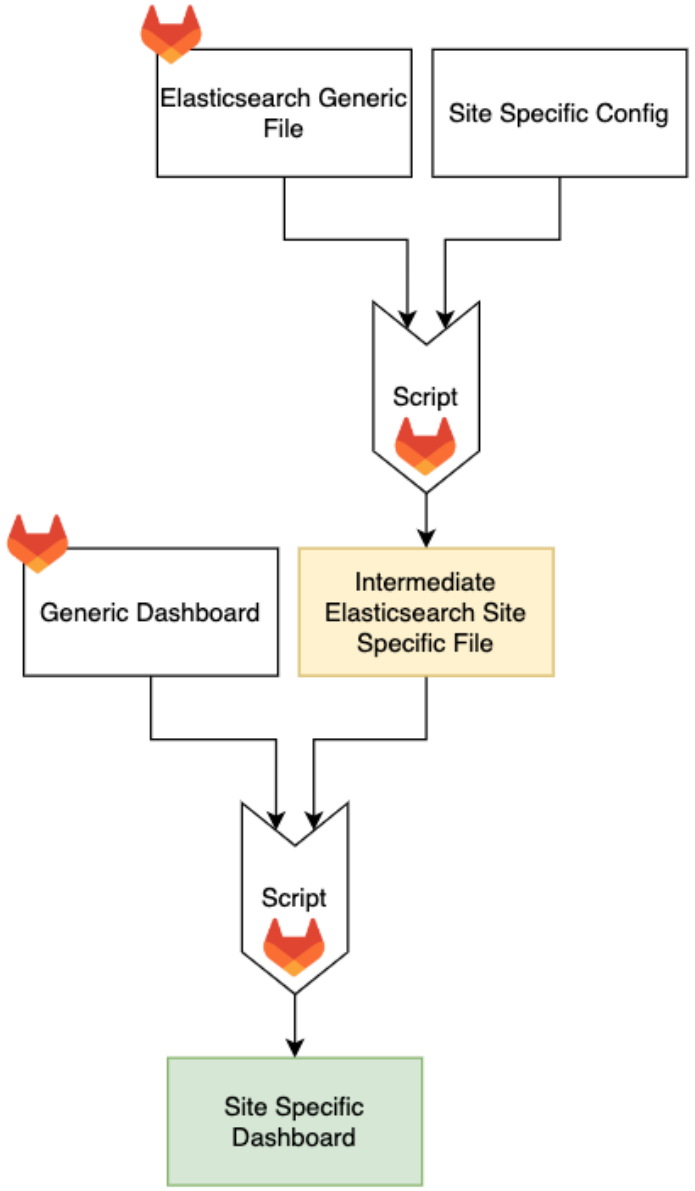
Generic Dashboard Config

```
"templating": {
 "list": [
   {
     "current": {
       "selected": false
     },
     "hide": 2,
     "includeAll": false,
     "multi": false,
     "name": "node_info",
     "options": [],
     "refresh": 1,
     "skipUrlSync": false,
     "type": "datasource"
   }
 ]
}
```

Site Specific Dashboard JSON Model

```
"templating": {
 "list": [
  {
    "current": {
      "selected": false,
      "text": "frontier.node-info",
      "value": "frontier.node-info"
    },
    "hide": 2,
    "includeAll": false,
    "multi": false,
    "name": "node_info",
    "options": [],
    "query": "elasticsearch",
    "refresh": 1,
    "regex": "/^frontier\\.node-info$/",
    "skipUrlSync": false,
    "type": "datasource"
  }
 ]
}
```

Workflow Summary



How to Use for Elasticsearch Datasources

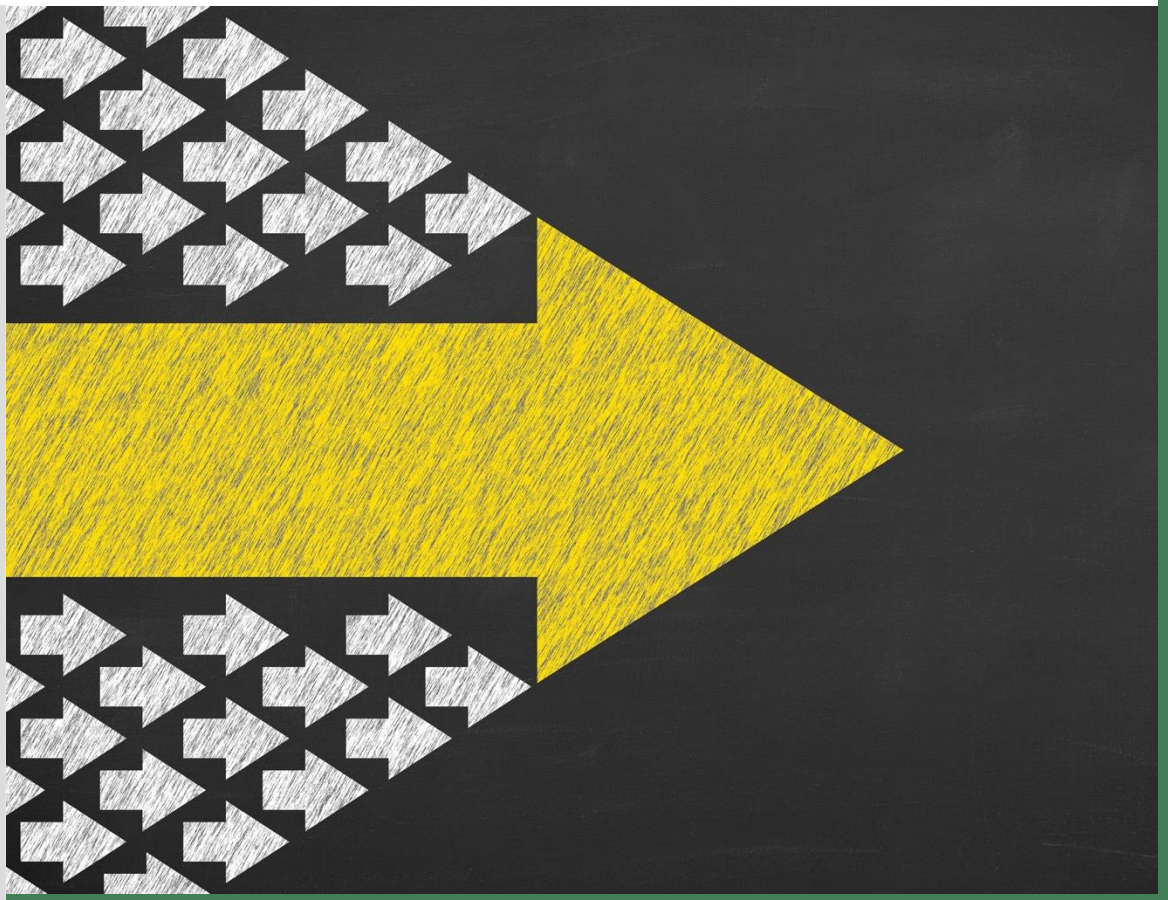
Create *your own Site-Specific Config file*

Call the **generator script** with the Elasticsearch Generic Config and your Site Specific Config

https://code.ornl.gov/oda-dashboards/oda-shared-dashboards.git.

Future Work

- Standardized HPC telemetry **schema**
- Implement and **test methods** at other HPC sites
- Improve **usability** with documentation, scripting, API access, broader backend database support
- Improve code base through Open Source **community contribution**



Thank you!

Tim Osborne: osbornetd@ornl.gov

Rachel Palumbo: palumborl@ornl.gov

Leah Huk: hukln@ornl.gov

Ryan Adamson: adamsonrm@ornl.gov

Rob Jones: jonesjr@ornl.gov

Corwin Lester: lestercp@ornl.gov

Acknowledgements:
This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.