Automatic run restarts

Belle II

+

Takuto KUNIGO

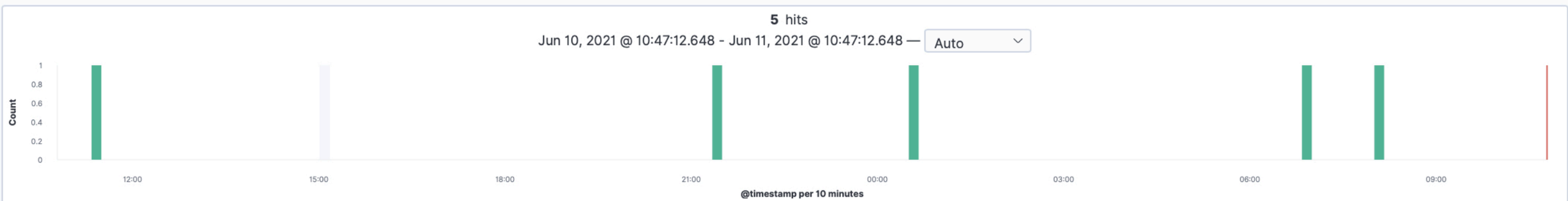
June 11, 2021

OTF meeting

KEK

Activated

Finally the auto run-restart mode activated!



Time	ExpNumber	RunNumber	Reason
Jun 11, 2021 @ 08:00:02.992	18	2,229	Beam abort
Jun 11, 2021 @ 06:51:36.498	18	2,228	Beam abort
Jun 11, 2021 @ 00:37:23.435	18	2,226	TTD busy
Jun 10, 2021 @ 21:20:31.958	18	2,224	-
Jun 10, 2021 @ 11:20:04.348	18	2,215	-

➡ disabled

However, further tuning and debugging needed

Automatic run-restarts

Current scheme to fix problems on our data-taking:

➡ Try SALS as soon as possible if SALS cannot fix the problem, contact the relevant on-call experts

- Sound alarm helps the CR shifters to take prompt action
- Automatic run restarts

Categories

1. Automatic restart for “inevitable” cases

- e.g. in case of the 8hour limit
- Automatic run stop at beam aborts, and/or automatic run start when HV permission granted can be categorised in this category

2. 1st trial of SALS

- i.e. 1st trial of SALS is requested/required most of the cases, we may trigger 1st trial of SALS automatically.
- However, we should not perform SALS against some errors which are not recoverable by SALS

Implementation/Flowchart

Needed info

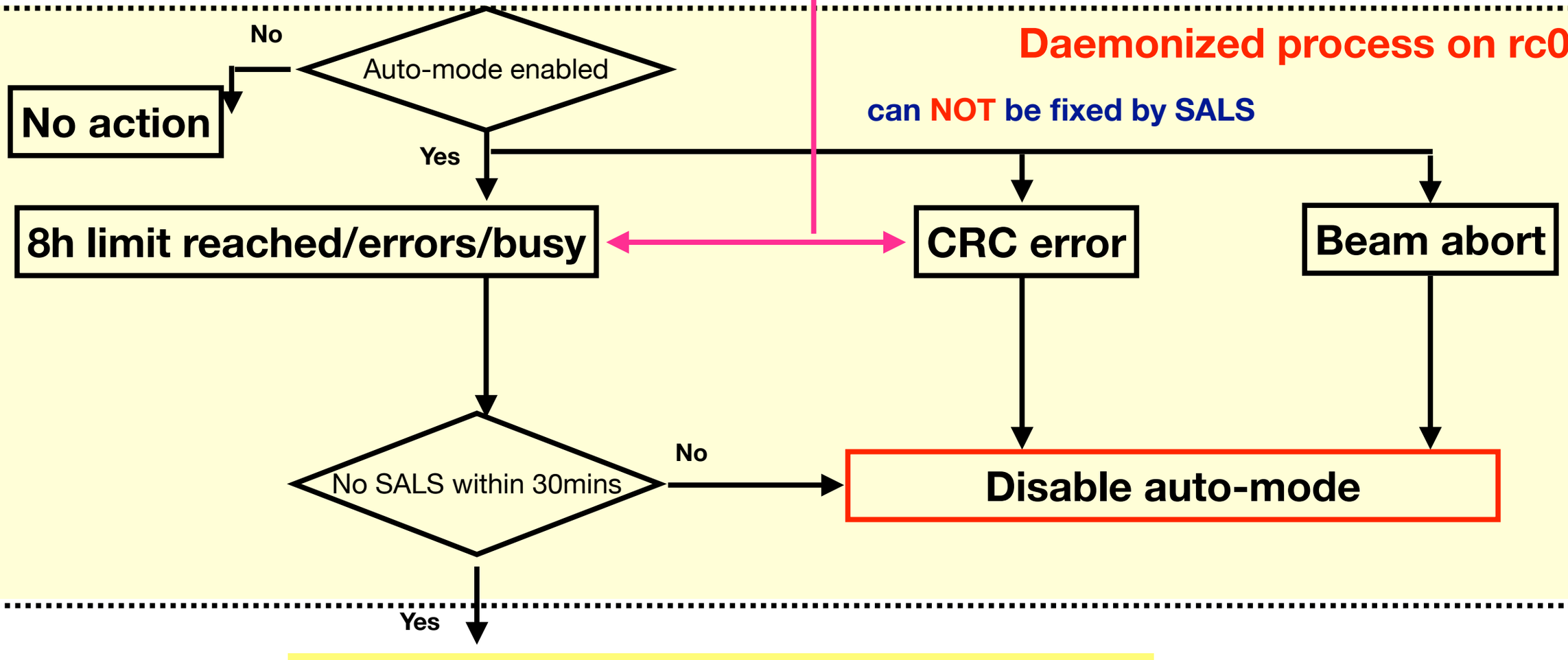
- Log messages
- PC metrics
- EPICS PVs
- nsm2 values
- Beam abort log

Checker

- Monitoring processes
- e.g Elasticsearch + Elastalert

Daemonized process on rc01

can NOT be fixed by SALS

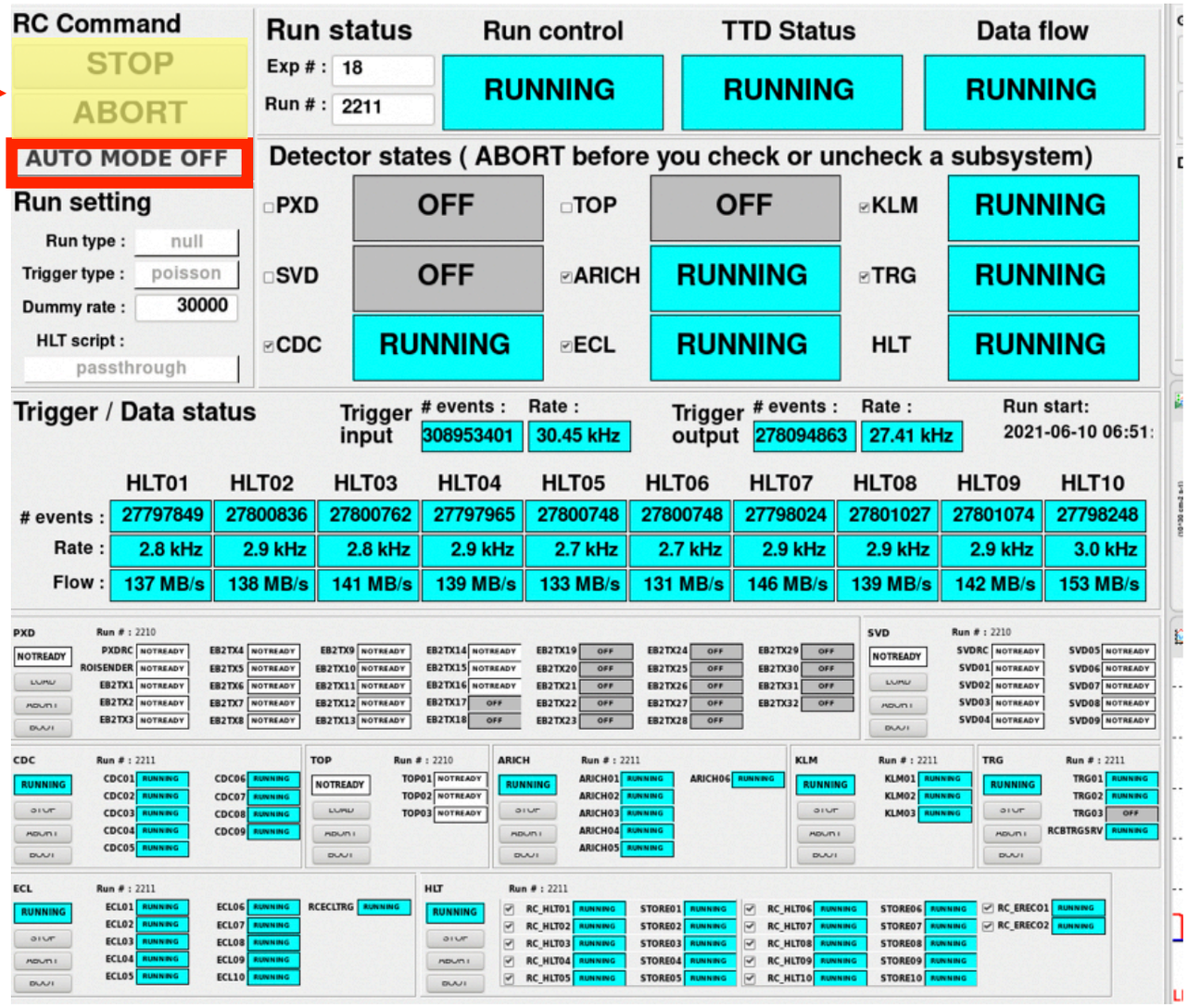


Automatic SALS/ALS/SS

From the CR shifters…

When auto mode is enabled, STOP/ABORT/LOAD/START buttons are un-clickable

B2_alarm:AUTORESTART:enable



Auto mode is enabled

- No sound alert
- RocketChat automatic message **for notification** +
  “Stop reason” in the elog entries filled (automatically) **for logging**

SALS and SS functions

- SALS and SS functions defined
- confirmed that these functions work fine

```python
#!/usr/bin/env python3
import time
import datetime
import epics
import nsm2

def SALS(timeout_start, timeout_load, timeout_abort, timeout_stop):
    nsm2.init('AUTORESTARTS', 9020)

    nsm2.send('RUNCONTROL', 'RC_STOP')
    time_send = datetime.datetime.now()
    while True:
        state = nsm2.vget('RUNCONTROL', 'rcstate')
        time.sleep(1) # remove?
        duration = (datetime.datetime.now() - time_send).total_seconds()
        if duration > timeout_stop:
            raise Exception('stuck at STOPPING')
        if state == 'READY':
            break
        elif state == "ERROR" or state == "FATAL":
            raise Exception('RC state is ERROR or FATAL')

    nsm2.send('RUNCONTROL', 'RC_ABORT')
    time_send = datetime.datetime.now()
    while True:
        state = nsm2.vget('RUNCONTROL', 'rcstate')
        time.sleep(1) # remove?
        duration = (datetime.datetime.now() - time_send).total_seconds()
        if duration > timeout_abort:
            raise Exception('stuck at ABORTING')
        if state == 'NOTREADY':
                break
        elif state == "ERROR" or state == "FATAL":
            raise Exception('RC state is ERROR or FATAL')
```

Unrecoverable errors

```yaml
Timeout:
  START: 300
  LOAD: 300
  ABORT: 300
  STOP: 300
UnrecoverableErrors:
  - '/home/group/b2daq/tkunigo/elk-input-scripts-temp/elastalert/rules/DAQ/crc_errors.yaml'
```

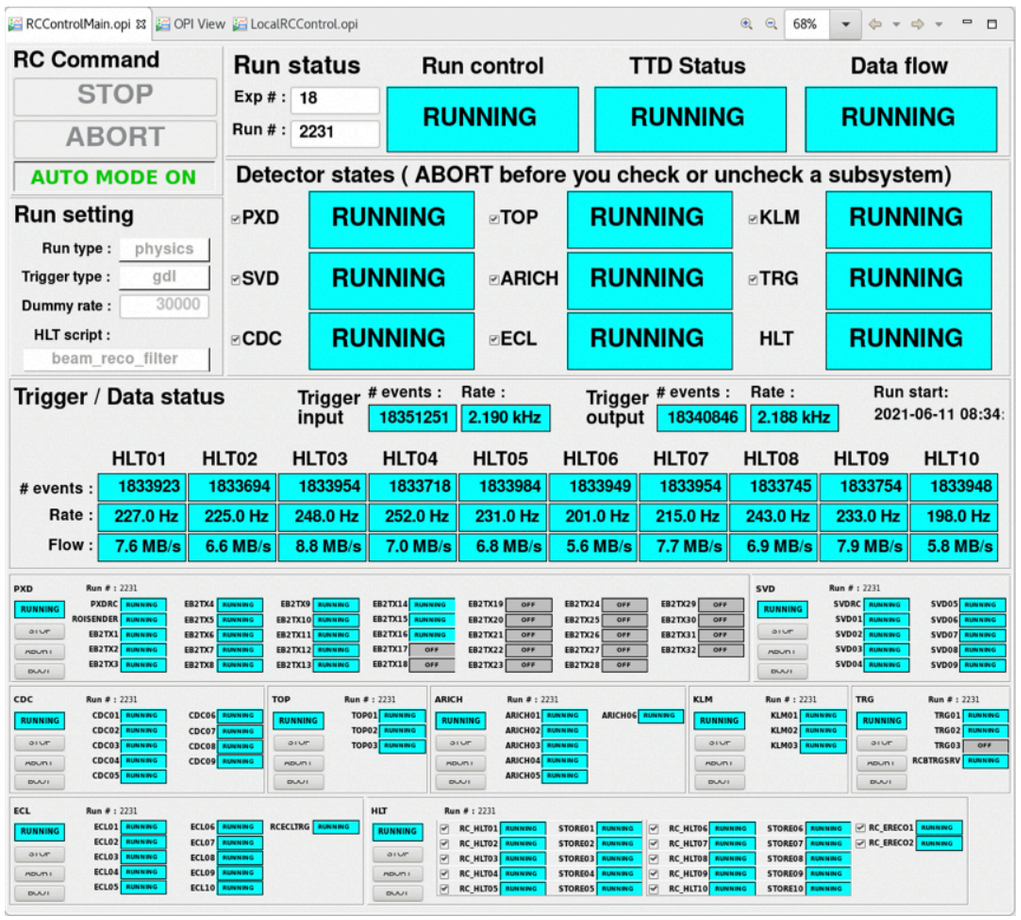
This list need to be updated
At least, I'll add the errors covered by the recover GUI

```python
# If error or busy
elif (rc_state=="ERROR") or (rc_state=="FATAL") or (error>0.0) or (busy>0.0):
    list_SALS = check_sliding_window()
    n_SALS    = len(list_SALS)

    # Unrecoverable errors
    UnrecoverableErrors = config_data['UnrecoverableErrors']
    unrecoverable = False
    for this_alert_rule in UnrecoverableErrors:
        if elasticsearch_query.query(this_alert_rule):
            unrecoverable = True
            break
    if unrecoverable:
        # disable the auto-mode when unrecoverable errors detected
        epics.caput(config_data['AutoRestartFlag'], 0.0)
    elif n_SALS<1:
        try:
            SALS(timeout_stop=config_data['Timeout']['STOP'], timeout_abort=config_data['Timeout']
        except Exception as e:
```

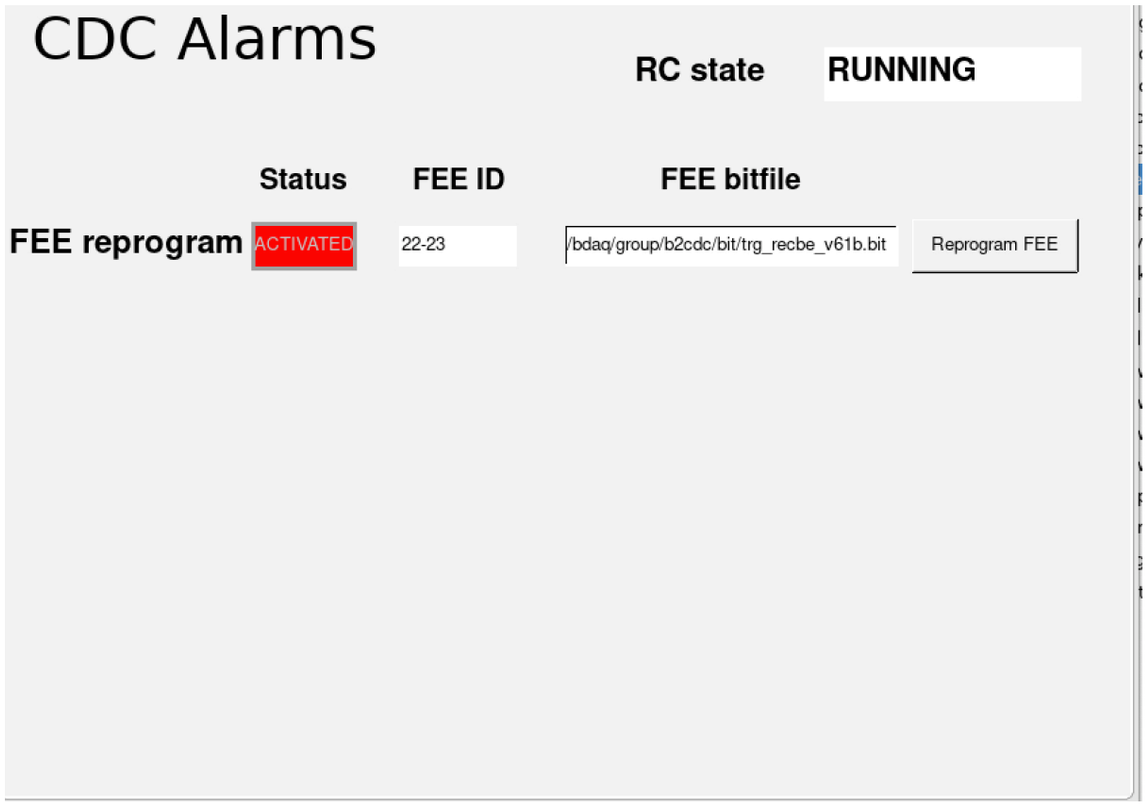
For future

Recoverable errors will be fixed by auto- SALS



+

“Unrecoverable errors” can be solved by the recovery GUI
➡ switch to automatic recovery



TOP and ARICH are joining

(Pseudo) automatic data-taking

This is not “Pie in the sky”

Backup

Demonstration

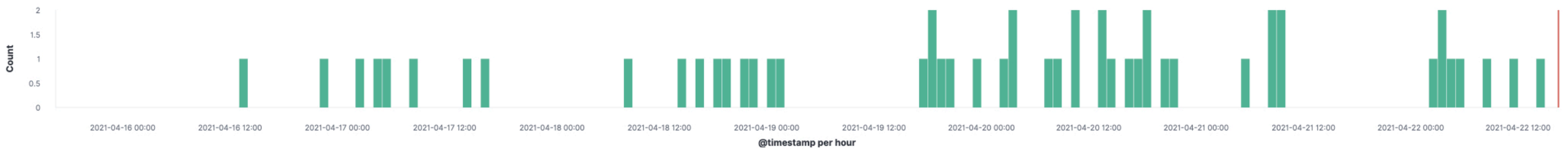
- **Git repo, link**
  - **config.yaml**

Controlled by supervisor b2daq@rc01
$ ~/.local/bin/supervisorctl [start/stop/restart] auto_restartd

```yaml
RCState: 'DAQ:RC:State'
RunType: 'DAQ:RC:RunType'
ExpNumber: 'DAQ:RC:ExpNumber'
RunNumber: 'DAQ:RC:RunNumber'
HVState: 'B2_nsm:get:HVMASTER:hvstate'
HVPermission: 'B2_MDI:HV_PERMISSION'
AutoRestartFlag: 'test:B2_alarm:AUTORESTART:enable'
SKB:
  Abort: 'B2_alarm:beam_abort'
TTD:
  tStart: 'DAQ:TTD:tStart'
  ERROR: 'B2_alarm:TTD:ftstate:ERROR'
  BUSY: 'B2_alarm:TTD:ftstate:BUSY'
NotRecoverableErrors:
  CRC errors: 'B2_alarm:DAQ:CRCErrors'
```

To be updated

Properly detected the events at which we should try automatic run restarts



Time	ExpNumber	RunNumber
Apr 22, 2021 @ 14:09:19.111	18	570
Apr 22, 2021 @ 11:14:45.081	18	561
Apr 22, 2021 @ 08:59:08.426	18	556
Apr 22, 2021 @ 05:36:26	18	554
Apr 22, 2021 @ 04:36:49.515	18	553
Apr 22, 2021 @ 03:48:10.917	18	550
Apr 22, 2021 @ 03:18:09.821	18	550