DAQ Trigger framework and appfwk

Pierre Lasorak

Introduction

- Following discussion in the last DataSelection meeting and Marco’s talk, I implemented simple trigger algorithms in appfwk
- Some of this is already outdated (discussions on Friday)
- Did the simplest thing I could think of:
    - Define trigger “objects”:
        - TriggerPrimitive
        - TriggerCandidate
        - TriggerDecision
            - Missing “TriggerCluster” or whatever we decide to call it
        - Define algorithms:
            - Maker class for each of the object

DuneTriggerAlgs

- Created an interface with appfwk
    - Each algorithm is it’s own DAQProcess
    - Defined queues etc

DAQDuneTriggers

Trigger Primitive

```
#pragma once
#include <cstdint>

namespace DuneTriggerAlgs {
  struct TriggerPrimitive {
    int64_t  time_start          = {0};
    int64_t  time_peak           = {0};
    int32_t  time_over_threshold = {0};
    uint32_t channel             = {0};
    uint16_t adc_integral        = {0};
    uint16_t adc_peak            = {0};
    uint32_t detid               = {0};
    uint32_t flag                = {0};
  };
}
```

- Very similar (identical?) to what is in PTMP.
- Hopefully that can accommodate for optical TPs?

Trigger Candidate

```
#pragma once
#include <cstdint>

namespace DuneTriggerAlgs {
  struct TriggerCandidate {
    int64_t  time_start    = {0};
    int64_t  time_end      = {0};
    int64_t  time_peak     = {0};
    uint32_t ntps          = {0};
    uint32_t channel_start = {0};
    uint32_t channel_end   = {0};
    uint32_t channel_peak  = {0};
    uint16_t adc_integral  = {0};
    uint16_t adc_peak      = {0};
    uint32_t detid         = {0};
    uint32_t flags         = {0};
  };
}
```

Trigger Decision

```
#pragma once
#include <cstdint>

namespace DuneTriggerAlgs {
  struct TriggerDecision {
    int64_t  time_start     = {0};
    int64_t  time_end       = {0};
    int64_t  time_triggered = {0};
    uint32_t detid          = {0};
    uint32_t flag           = {0};
  };
}
```

Algorithm “Maker”

```
#pragma once
#include "dune-trigger-algs/TriggerPrimitive.hh"
#include "dune-trigger-algs/TriggerCandidate.hh"
#include <vector>
#include <string>

namespace DuneTriggerAlgs {

  class TriggerCandidateMaker {
  public:
    virtual void operator()(const TriggerPrimitive& input_tp,
                            std::vector<TriggerCandidate>& output_tc) = 0;
    virtual void flush(std::vector<TriggerCandidate>&) {};
  };

}
```

- One for each object (TriggerPrimitiveMaker, TriggerCandidateMaker, TriggerDecisionMaker)
- operator() pure virtual function → that’s where all the algorithmic part happens
- flush is maybe not needed…
  - At the end of the run, maybe somebody is interested in ill-formed clusters which had to be truncated?

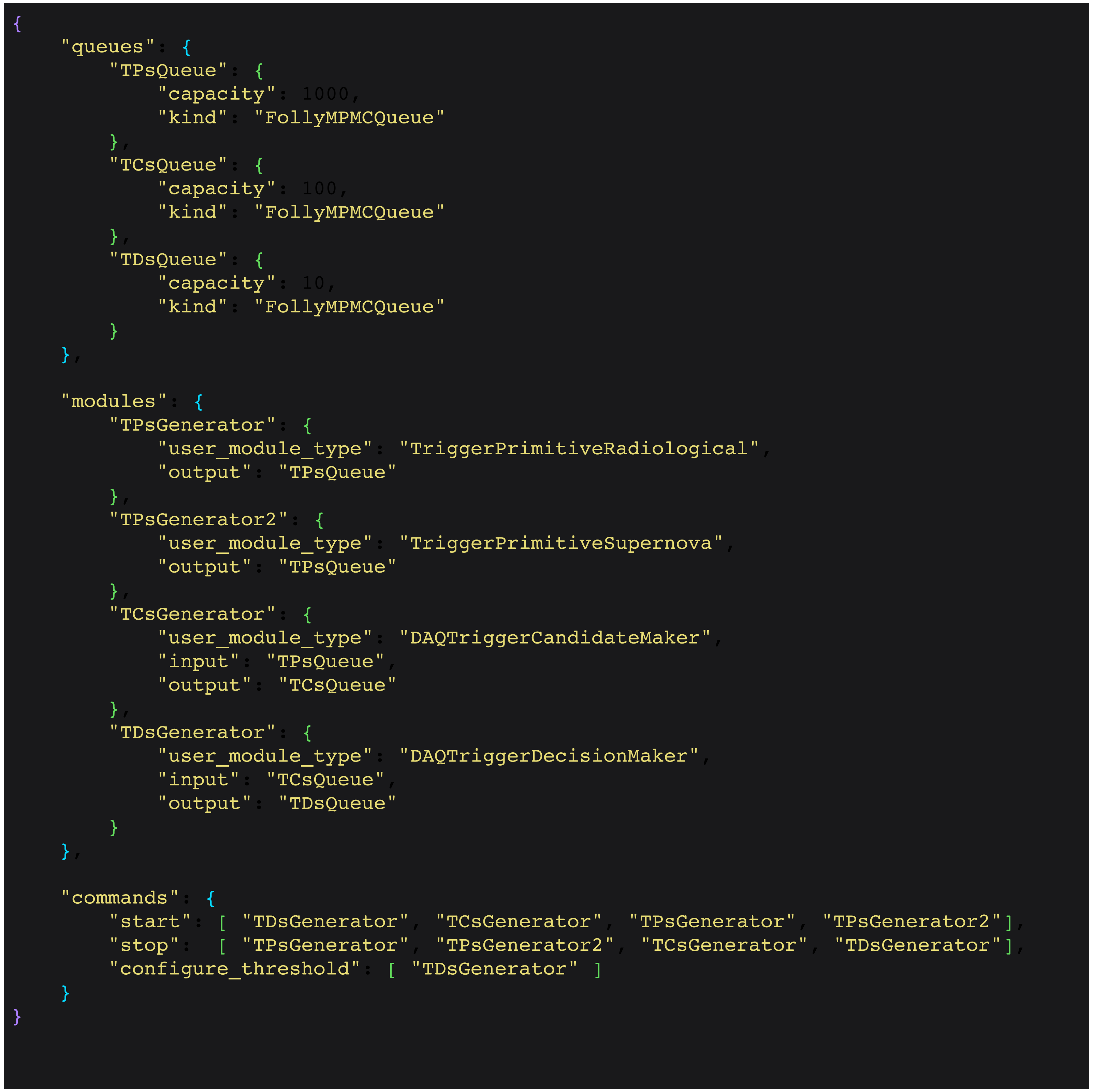
Appfwk interface

- Where the DAQProcesses are implemented and calling these operator() functions
- Example class:

```
class DAQTriggerCandidateMaker: public dunedaq::appfwk::DAQModule,
                                       DuneTriggerAlgs::TriggerCandidateMakerSupernova
```

- Fills and consumes queues of Trigger objects
  - Simplest thing I could think of (most of it is just copy-paste from listrev)
  - Obviously, all of this has to run on the same host since there isn't any message protocol implemented in the appfwk (or at least I didn't know about it)
- Holds and parses all the configurations of the algorithms previously defined.
  - For example, in case of Supernova trigger decision maker, we might change the threshold in number of clusters at which the SN trigger is emitted
  - Implemented as simple `std::atomic<int>` in TriggerCandidateMaker, the DAQProcess is in charge to update it (i.e. reconfiguration can happen without a "stop and start" sequence)
    - Didn't quite make it happen for this talk

simplest_trigger.json



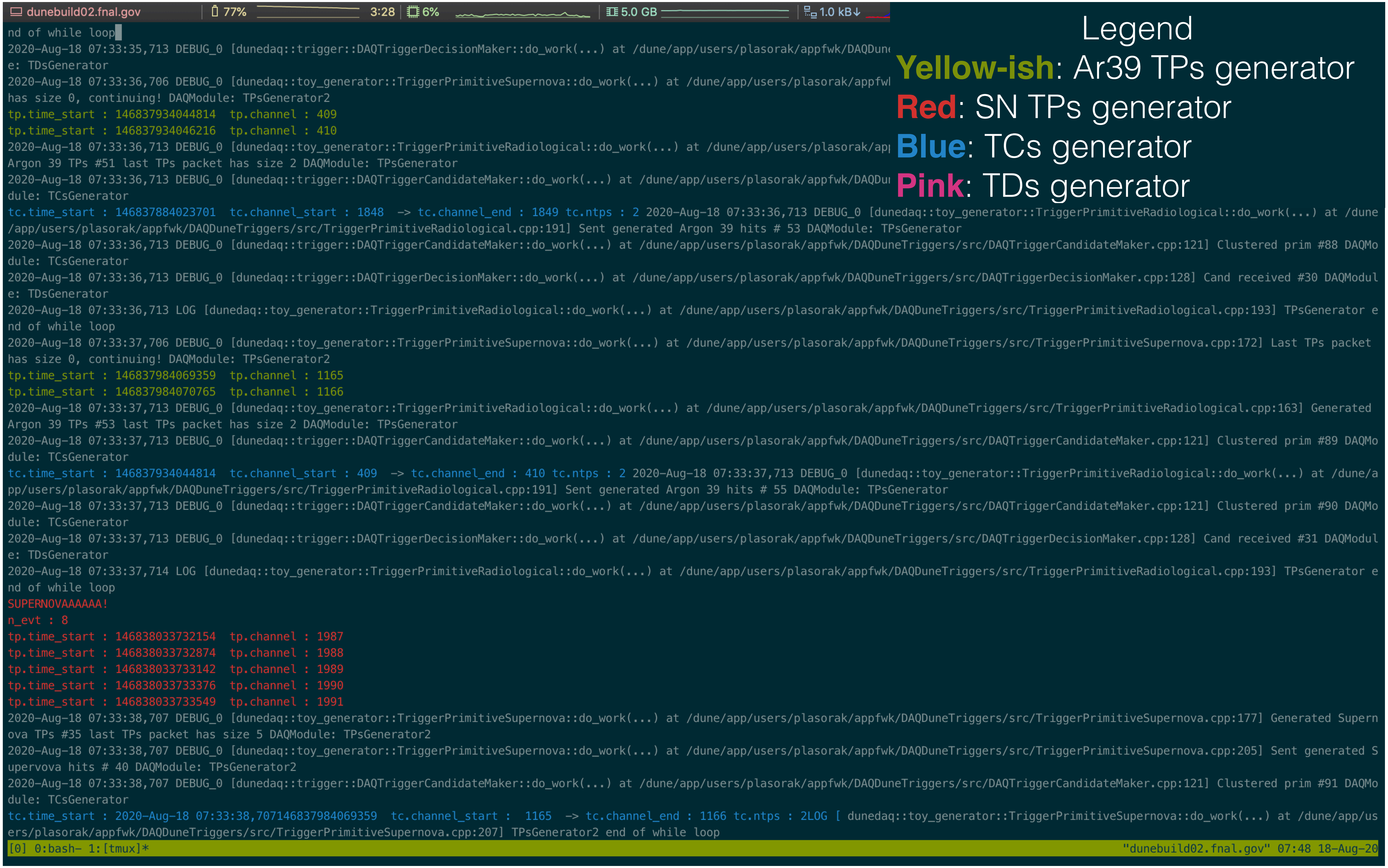
```
{
    "queues": {
        "TPsQueue": {
            "capacity": 1000,
            "kind": "FollyMPMCQueue"
        },
        "TCsQueue": {
            "capacity": 100,
            "kind": "FollyMPMCQueue"
        },
        "TDsQueue": {
            "capacity": 10,
            "kind": "FollyMPMCQueue"
        }
    },

    "modules": {
        "TPsGenerator": {
            "user_module_type": "TriggerPrimitiveRadiological",
            "output": "TPsQueue"
        },
        "TPsGenerator2": {
            "user_module_type": "TriggerPrimitiveSupernova",
            "output": "TPsQueue"
        },
        "TCsGenerator": {
            "user_module_type": "DAQTriggerCandidateMaker",
            "input": "TPsQueue",
            "output": "TCsQueue"
        },
        "TDsGenerator": {
            "user_module_type": "DAQTriggerDecisionMaker",
            "input": "TCsQueue",
            "output": "TDsQueue"
        }
    },

    "commands": {
        "start": [ "TDsGenerator", "TCsGenerator", "TPsGenerator", "TPsGenerator2"],
        "stop":  [ "TPsGenerator", "TPsGenerator2", "TCsGenerator", "TDsGenerator"],
        "configure_threshold": [ "TDsGenerator" ]
    }
}
```

- Stores everything in MPMCQueues (probably not the best choice…)
- Generators:
    - TCsGenerator consumes TPs
    - TDsGenerator consumes TCs
    - TPsGenerator should consumes something representative of the raw data
        - Right now, just generates random TPs (Argon 39 and Supernova)
- Of course this is cheating
    - Although there are 2 threads generating TPs, they are nicely “time ordered”, there are some assumptions during the formation of TCs would make it break if that wasn’t the case
- Would need equivalent of a ptmp’s “TPwindow” to make it more realistic

Running it



Legend

Yellow-ish: Ar39 TPs generator

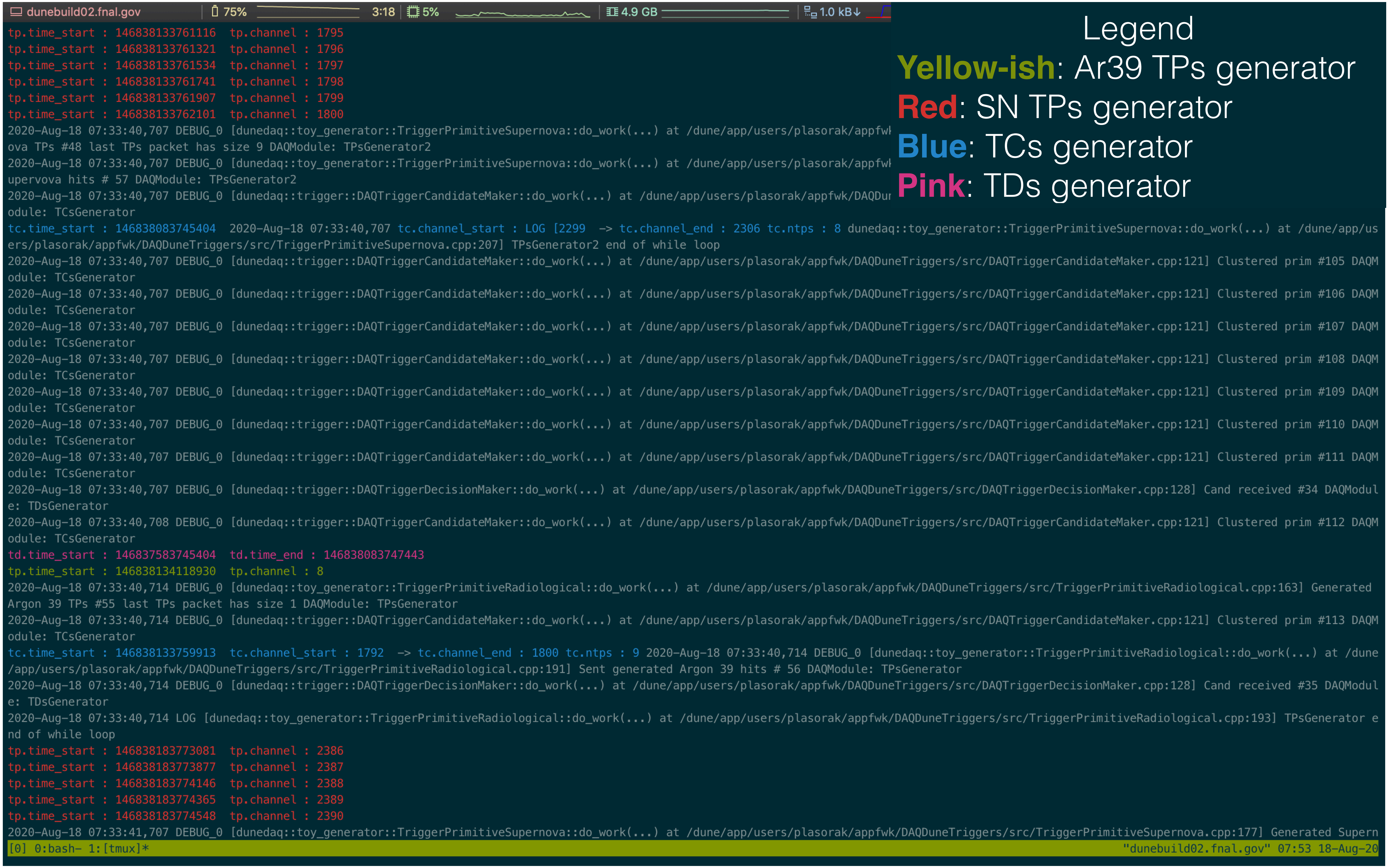
Red: SN TPs generator

Blue: TCs generator

Pink: TDs generator

```
dunebuild02.fnal.gov   77%   3:28   6%   5.0 GB   1.0 kB↓
nd of while loop
2020-Aug-18 07:33:35,713 DEBUG_0 [dunedaq::trigger::DAQTriggerDecisionMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDun
e: TDsGenerator
2020-Aug-18 07:33:36,706 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwl
has size 0, continuing! DAQModule: TPsGenerator2
tp.time_start : 146837934044814  tp.channel : 409
tp.time_start : 146837934046216  tp.channel : 410
2020-Aug-18 07:33:36,713 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/app/users/plasorak/ap
Argon 39 TPs #51 last TPs packet has size 2 DAQModule: TPsGenerator
2020-Aug-18 07:33:36,713 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDu
dule: TCsGenerator
tc.time_start : 146837884023701  tc.channel_start : 1848  -> tc.channel_end : 1849 tc.ntps : 2 2020-Aug-18 07:33:36,713 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune
/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:191] Sent generated Argon 39 hits # 53 DAQModule: TPsGenerator
2020-Aug-18 07:33:36,713 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #88 DAQMo
dule: TCsGenerator
2020-Aug-18 07:33:36,713 DEBUG_0 [dunedaq::trigger::DAQTriggerDecisionMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerDecisionMaker.cpp:128] Cand received #30 DAQModul
e: TDsGenerator
2020-Aug-18 07:33:36,713 LOG [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:193] TPsGenerator e
nd of while loop
2020-Aug-18 07:33:37,706 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveSupernova.cpp:172] Last TPs packet
has size 0, continuing! DAQModule: TPsGenerator2
tp.time_start : 146837984069359  tp.channel : 1165
tp.time_start : 146837984070765  tp.channel : 1166
2020-Aug-18 07:33:37,713 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:163] Generated
Argon 39 TPs #53 last TPs packet has size 2 DAQModule: TPsGenerator
2020-Aug-18 07:33:37,713 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #89 DAQMo
dule: TCsGenerator
tc.time_start : 146837934044814  tc.channel_start : 409  -> tc.channel_end : 410 tc.ntps : 2 2020-Aug-18 07:33:37,713 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/a
pp/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:191] Sent generated Argon 39 hits # 55 DAQModule: TPsGenerator
2020-Aug-18 07:33:37,713 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #90 DAQMo
dule: TCsGenerator
2020-Aug-18 07:33:37,713 DEBUG_0 [dunedaq::trigger::DAQTriggerDecisionMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerDecisionMaker.cpp:128] Cand received #31 DAQModul
e: TDsGenerator
2020-Aug-18 07:33:37,714 LOG [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:193] TPsGenerator e
nd of while loop
SUPERNOVAAAAAAA!
n_evt : 8
tp.time_start : 146838033732154  tp.channel : 1987
tp.time_start : 146838033732874  tp.channel : 1988
tp.time_start : 146838033733142  tp.channel : 1989
tp.time_start : 146838033733376  tp.channel : 1990
tp.time_start : 146838033733549  tp.channel : 1991
2020-Aug-18 07:33:38,707 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveSupernova.cpp:177] Generated Supern
ova TPs #35 last TPs packet has size 5 DAQModule: TPsGenerator2
2020-Aug-18 07:33:38,707 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveSupernova.cpp:205] Sent generated S
upervova hits # 40 DAQModule: TPsGenerator2
2020-Aug-18 07:33:38,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #91 DAQMo
dule: TCsGenerator
tc.time_start : 2020-Aug-18 07:33:38,707146837984069359  tc.channel_start :  1165  -> tc.channel_end : 1166 tc.ntps : 2LOG [ dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/us
ers/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveSupernova.cpp:207] TPsGenerator2 end of while loop
[0] 0:bash- 1:[tmux]*                                                                        "dunebuild02.fnal.gov" 07:48 18-Aug-20
```

… and a bit further



```
dunebuild02.fnal.gov | 75% 3:18 | 5% | 4.9 GB | 1.0 kB↓
tp.time_start : 146838133761116  tp.channel : 1795
tp.time_start : 146838133761321  tp.channel : 1796
tp.time_start : 146838133761534  tp.channel : 1797
tp.time_start : 146838133761741  tp.channel : 1798
tp.time_start : 146838133761907  tp.channel : 1799
tp.time_start : 146838133762101  tp.channel : 1800
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwk
ova TPs #48 last TPs packet has size 9 DAQModule: TPsGenerator2
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwk
upervova hits # 57 DAQModule: TPsGenerator2
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDur
odule: TCsGenerator
tc.time_start : 146838083745404  2020-Aug-18 07:33:40,707 tc.channel_start : LOG [2299  -> tc.channel_end : 2306 tc.ntps : 8 dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/us
ers/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveSupernova.cpp:207] TPsGenerator2 end of while loop
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #105 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #106 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #107 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #108 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #109 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #110 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #111 DAQM
odule: TCsGenerator
2020-Aug-18 07:33:40,707 DEBUG_0 [dunedaq::trigger::DAQTriggerDecisionMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerDecisionMaker.cpp:128] Cand received #34 DAQModul
e: TDsGenerator
2020-Aug-18 07:33:40,708 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #112 DAQM
odule: TCsGenerator
td.time_start : 146837583745404  td.time_end : 146838083747443
tp.time_start : 146838134118930  tp.channel : 8
2020-Aug-18 07:33:40,714 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:163] Generated
Argon 39 TPs #55 last TPs packet has size 1 DAQModule: TPsGenerator
2020-Aug-18 07:33:40,714 DEBUG_0 [dunedaq::trigger::DAQTriggerCandidateMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerCandidateMaker.cpp:121] Clustered prim #113 DAQM
odule: TCsGenerator
tc.time_start : 146838133759913  tc.channel_start : 1792  -> tc.channel_end : 1800 tc.ntps : 9 2020-Aug-18 07:33:40,714 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune
/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:191] Sent generated Argon 39 hits # 56 DAQModule: TPsGenerator
2020-Aug-18 07:33:40,714 DEBUG_0 [dunedaq::trigger::DAQTriggerDecisionMaker::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/DAQTriggerDecisionMaker.cpp:128] Cand received #35 DAQModul
e: TDsGenerator
2020-Aug-18 07:33:40,714 LOG [dunedaq::toy_generator::TriggerPrimitiveRadiological::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveRadiological.cpp:193] TPsGenerator e
nd of while loop
tp.time_start : 146838183773081  tp.channel : 2386
tp.time_start : 146838183773877  tp.channel : 2387
tp.time_start : 146838183774146  tp.channel : 2388
tp.time_start : 146838183774365  tp.channel : 2389
tp.time_start : 146838183774548  tp.channel : 2390
2020-Aug-18 07:33:41,707 DEBUG_0 [dunedaq::toy_generator::TriggerPrimitiveSupernova::do_work(...) at /dune/app/users/plasorak/appfwk/DAQDuneTriggers/src/TriggerPrimitiveSupernova.cpp:177] Generated Supern
[0] 0:bash- 1:[tmux]*                                                                                          "dunebuild02.fnal.gov" 07:53 18-Aug-20
```

Legend

Yellow-ish: Ar39 TPs generator

Red: SN TPs generator

Blue: TCs generator

Pink: TDs generator

Conclusion

- Basic SN triggering scheme implemented in the appfwk
- Quite a bit of bricks are missing to make it usable in the real system, here are the one I can think of:
  - Messaging system in appfwk
  - TP window
  - This intermediate state between TCs and TDs
  - A better idea how to implement TP generator from raw data (no input queue for the TPs in my example)
- Where to go from here?
  - https://github.com/plasorak/DAQDuneTriggers
  - https://github.com/plasorak/DuneTriggerAlgs