

# Full 10kt Sim/SigProc Update

Haiwang Yu (BNL)

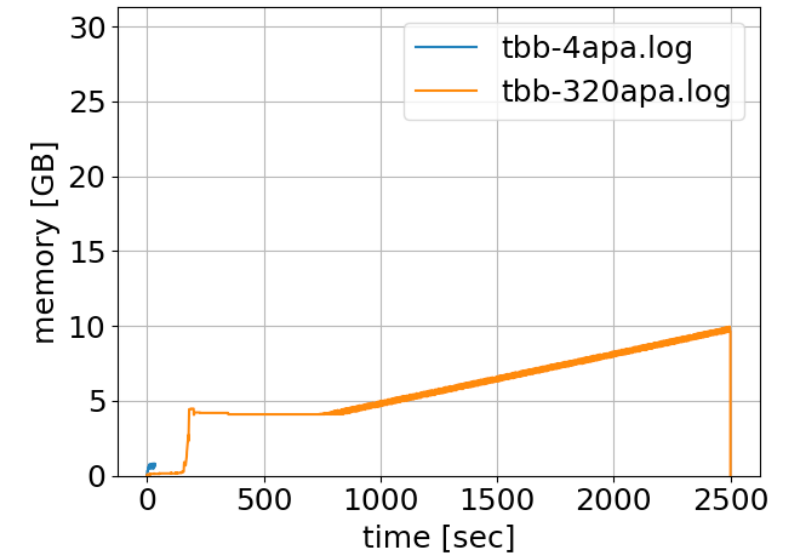
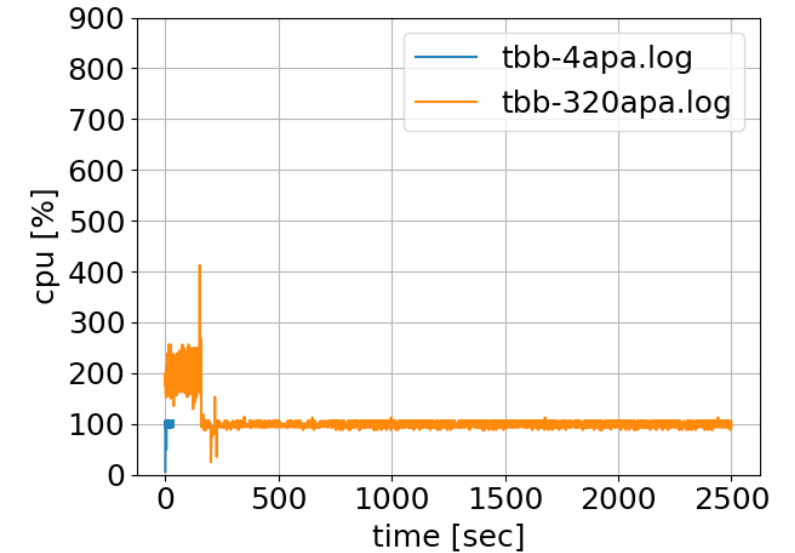
2023-11-06



@BrookhavenLab

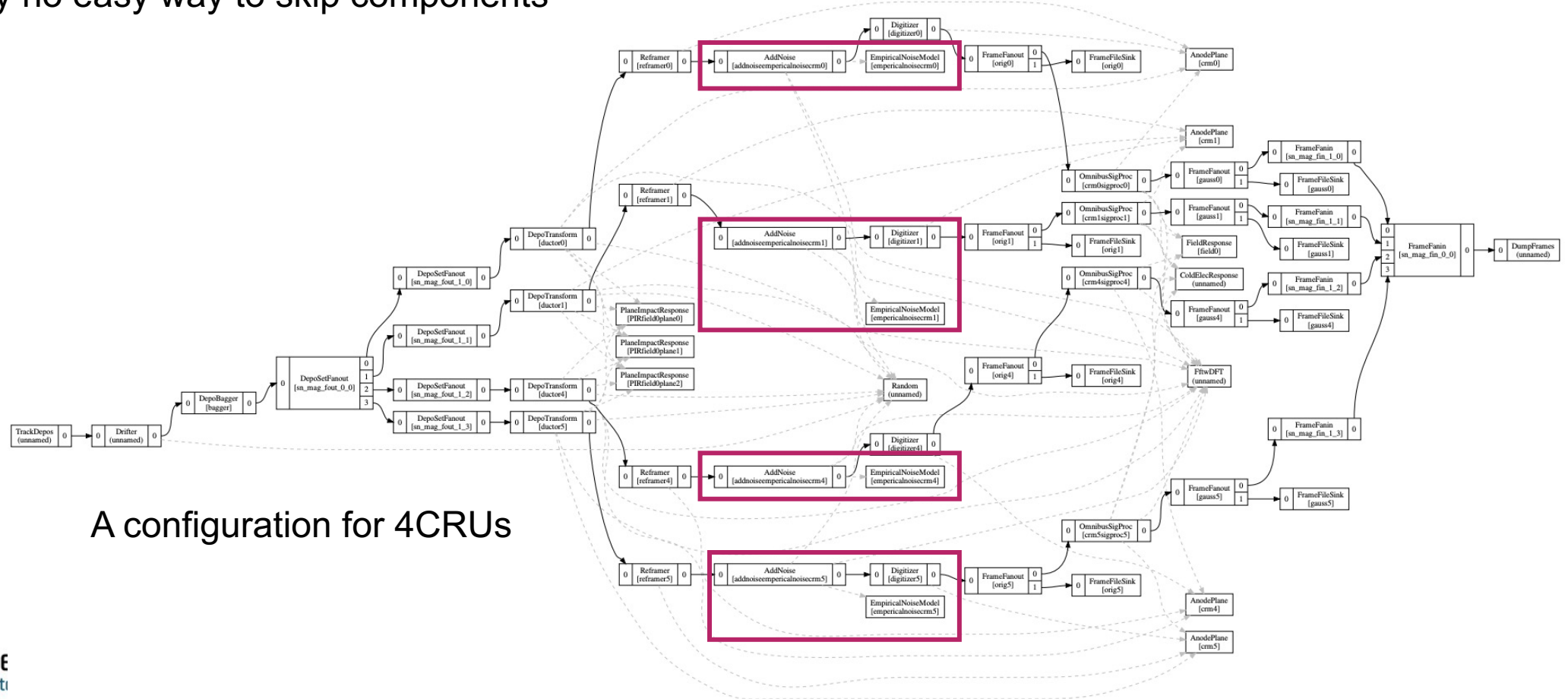
# Full 10kt initial test

- Initial testing showed in the 2023 Jan Collab. Meeting
  - <https://indico.fnal.gov/event/53965/contributions/257909/>
  - Memory/time could be reduced
- Some updates on memory reduction and plans on speed optimization today



# Wire-Cell Dataflow graph

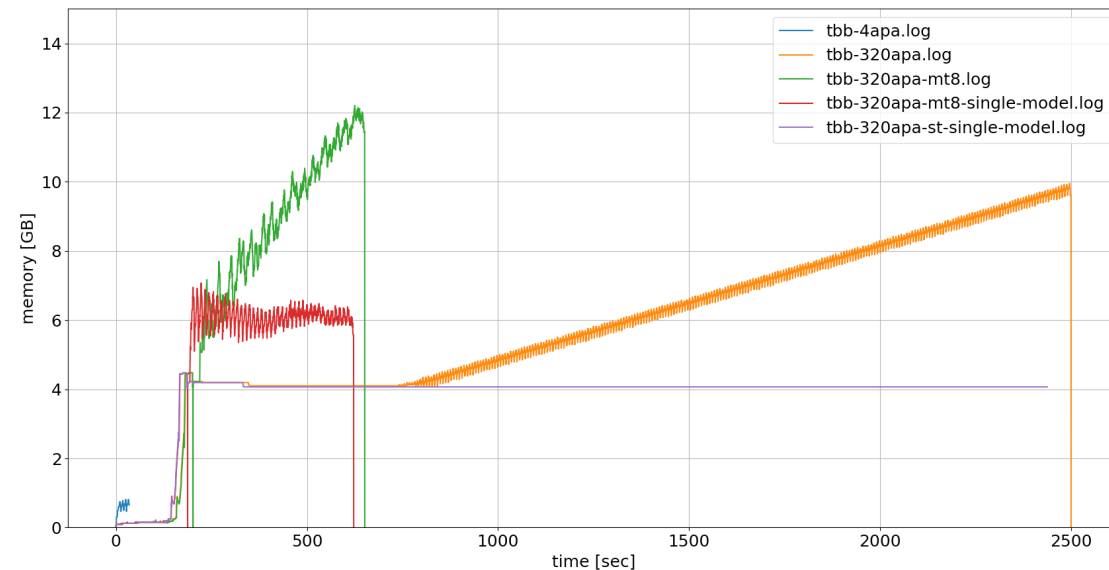
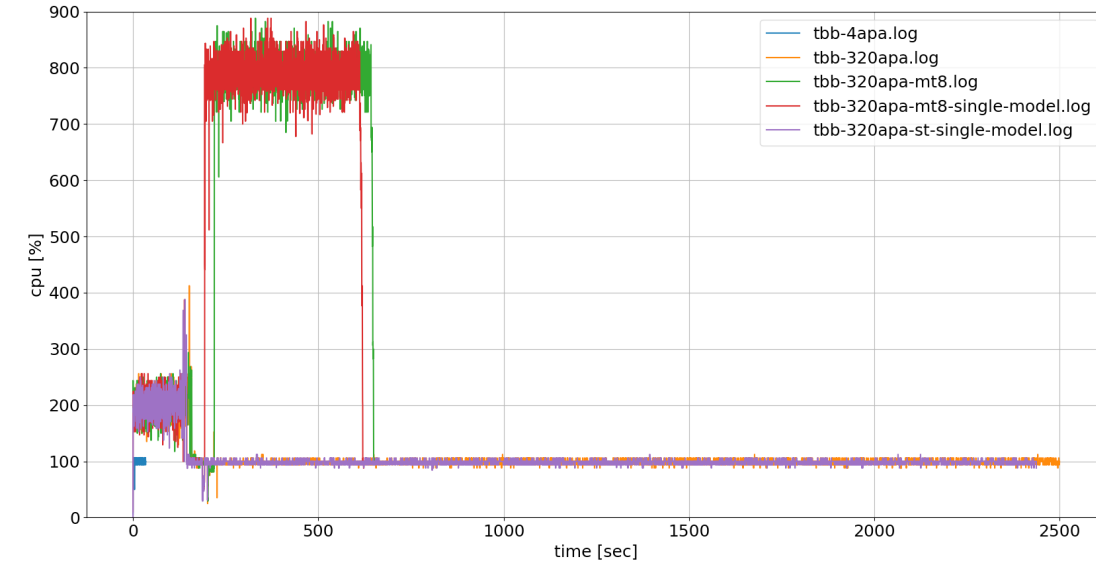
- Motivated by multi-threading/memory share needs
- Need per-APA class instance – per-APA memory increasing
  - components
  - services
- Currently no easy way to skip components



# Use one NoiseModel for all APAs

Some caching used in the NoiseModel service (profiling in backup slides)

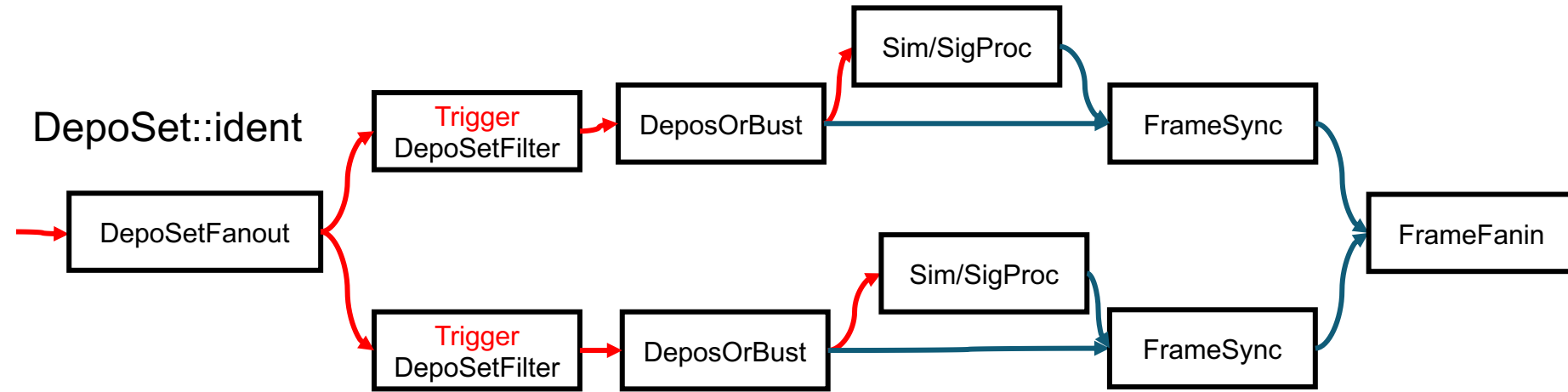
- currently every AddNoise component has its own NoiseModel service
- noise spectra determined by plane, wire-length, gain, shaping
  - → could be independent of APAs
  - → one NoiseModel for all APAs
- Need a lock to protect cache modification (service needs to be thread safe)
- The per-APA increasing part is gone if use this
  - ST: 2430 sec, ~4GB
  - 8-threads: 620 sec, ~6GB
    - 180 compiling jsonnet, 420 core processing



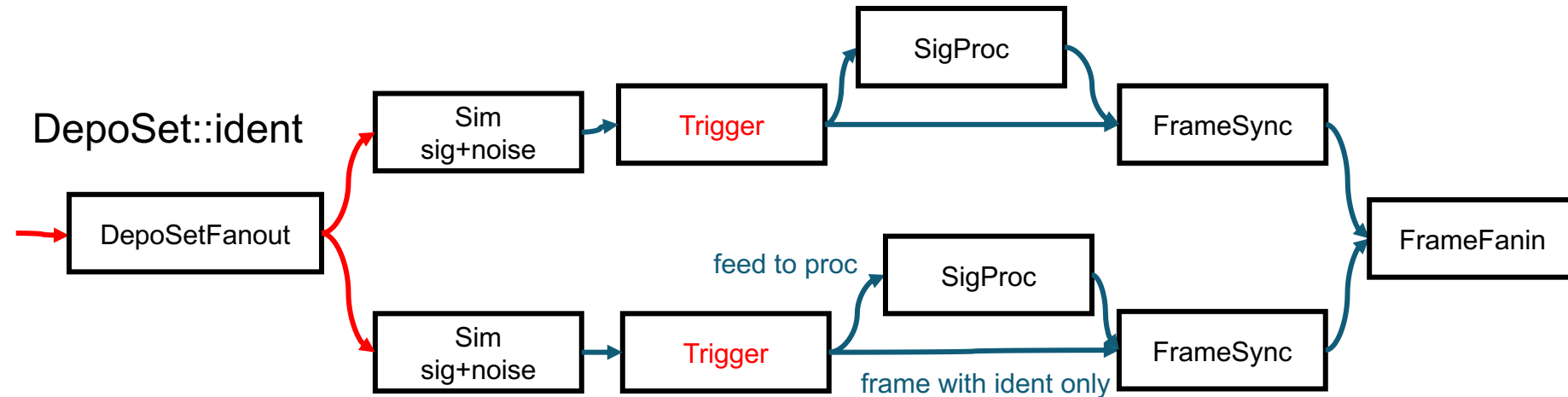
# Shortcuts to skip some processing

[issue #148](#)

Cheat trigger to skip both Sim/SigProc



- Sim all APAs
- Trigger alg. to determine whether to do SigProc



# Next

Validate the single noise model method

Finish the async nodes/skip path

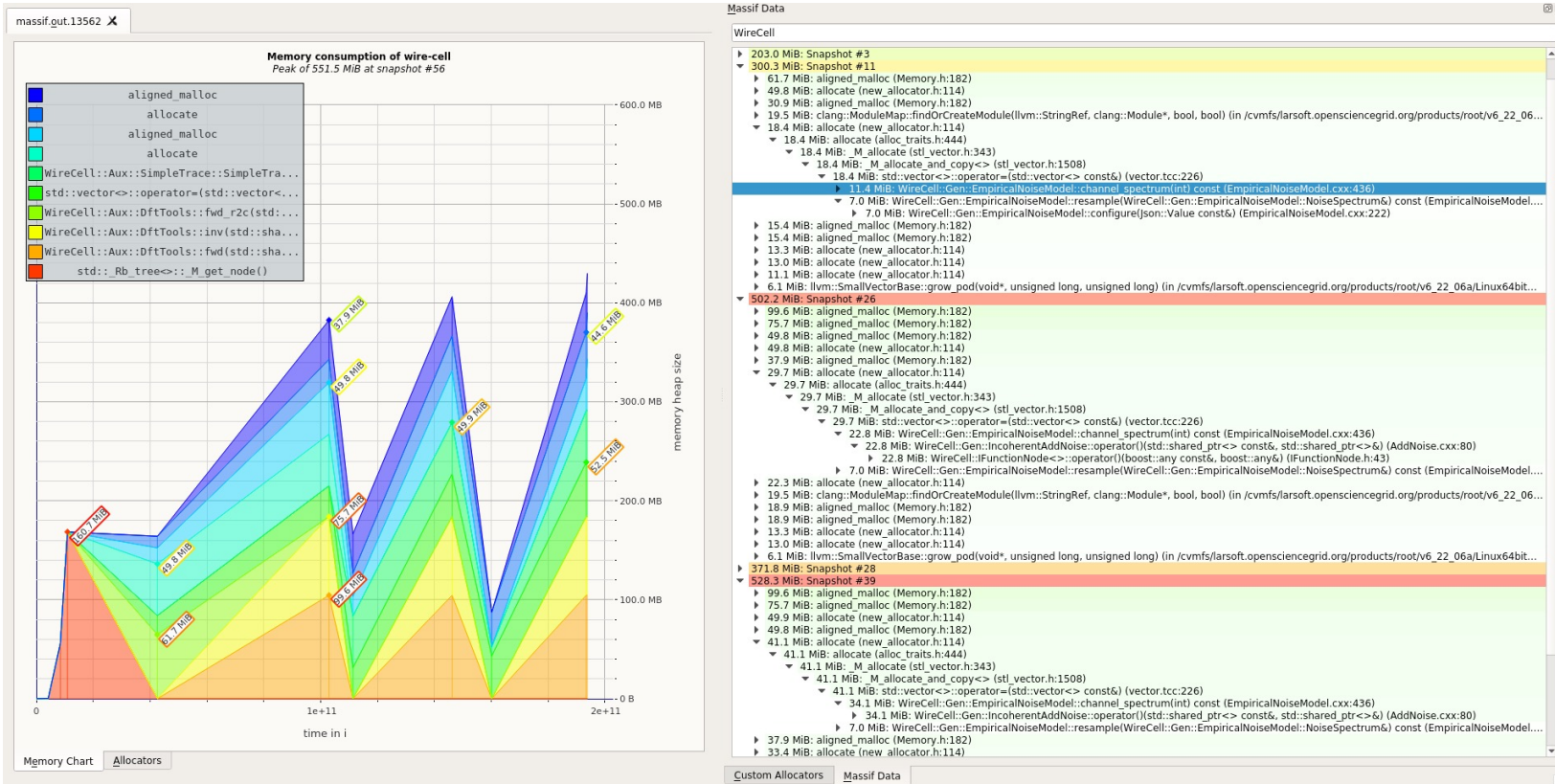
- <https://github.com/WireCell/wire-cell-toolkit/pull/169>

Further reduce the per-APA component memory

Randomness: neither engines in WCT is deterministic for now

- [issue 242](#)
- Easier to fix the pgrapher one (single thread only).
- Longer term to fix the TbbFlow (multi-thread).

# backup

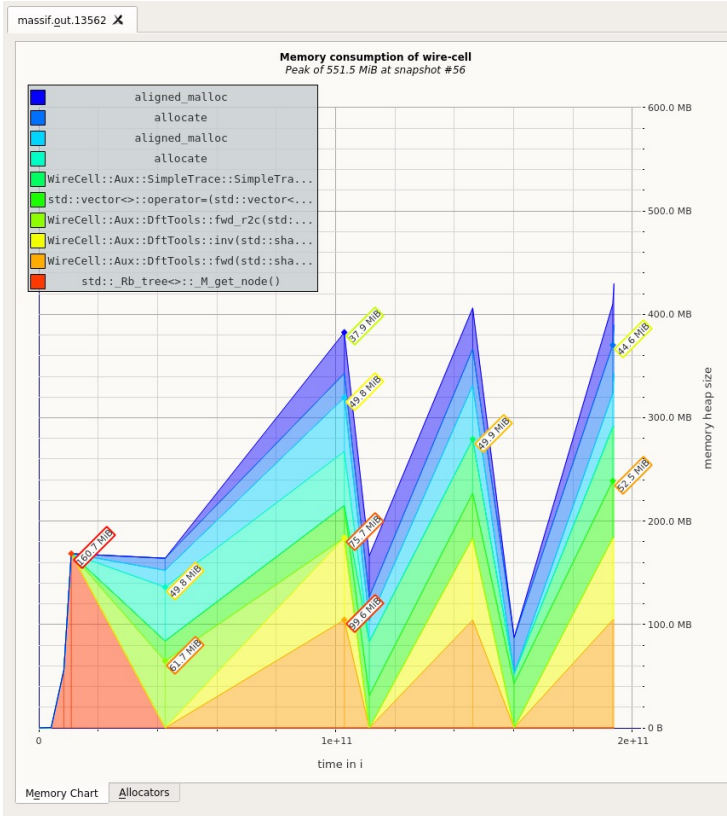


```

433     ... unsigned int packkey = pack();
434     ... m_channel_packkey_cache[chid] = packkey;
435     ... // m_channel_amp_cache.clear();
436     ... m_channel_amp_cache[packkey] = amp;
437     ... return channel_spectrum(chid);

```





### Massif Data

WireCell

```

▼ 29.7 MiB: std::vector<>::operator=(std::vector<> const&) (vector.tcc:226)
▼ 22.8 MiB: WireCell::Gen::EmpiricalNoiseModel::channel_spectrum(int) const (EmpiricalNoiseModel.cxx:436)
  ▼ 22.8 MiB: WireCell::Gen::incoherentAddNoise::operator()(std::shared_ptr<> const&, std::shared_ptr<> s) (AddNoise.cxx:80)
    ▼ 22.8 MiB: WireCell::IFunctionNode<>::operator()() const (IFunctionNode.h:43)
      ▼ 7.0 MiB: WireCell::Gen::EmpiricalNoiseModel::resample(WireCell::Gen::EmpiricalNoiseModel::NoiseSpectrum&) const (EmpiricalNoiseModel.cxx:276)
        ▼ 22.3 MiB: allocate (new_allocator.h:114)
          ▼ 22.3 MiB: allocate (alloc_traits.h:444)
            ▼ 22.3 MiB: M_allocate (stl_vector.h:343)
              ▼ 22.3 MiB: M_create_storage (stl_vector.h:358)
                ▼ 22.3 MiB: _Vector_base (stl_vector.h:302)
                  ▼ 22.3 MiB: vector (stl_vector.h:552)
                    ▼ 22.3 MiB: pair<> (stl_pair.h:342)
                      ▼ 22.3 MiB: WireCell::Gen::EmpiricalNoiseModel::interpolate_wire_length(int, double) const (EmpiricalNoiseModel.cxx:276)
                        ▼ 19.5 MiB: clang::ModuleMap::findOrCreateModule(Illvm::StringRef, clang::Module*, bool, bool) (in /cvfms/larsoft.opensciencgrid.org/products/root/v6_22_06...
                          ▼ 18.9 MiB: aligned_malloc (Memory.h:182)
                          ▼ 18.9 MiB: aligned_malloc (Memory.h:182)
                          ▼ 13.3 MiB: allocate (new_allocator.h:114)
                          ▼ 13.0 MiB: allocate (new_allocator.h:114)
                          ▼ 6.1 MiB: llvms::SmallVectorBase::grow_pod(void*, unsigned long, unsigned long) (in /cvfms/larsoft.opensciencgrid.org/products/root/v6_22_06a/Linux64bit...
                        ▼ 371.8 MiB: Snapshot #28
                        ▼ 528.3 MiB: Snapshot #39
                          ▼ 99.6 MiB: aligned_malloc (Memory.h:182)
                          ▼ 75.7 MiB: aligned_malloc (Memory.h:182)
                          ▼ 49.9 MiB: allocate (new_allocator.h:114)
                          ▼ 49.9 MiB: allocate (alloc_traits.h:444)
                            ▼ 49.9 MiB: M_allocate (stl_vector.h:343)
                              ▼ 49.9 MiB: M_create_storage (stl_vector.h:358)
                                ▼ 49.9 MiB: _Vector_base (stl_vector.h:302)
                                  ▼ 49.9 MiB: vector (stl_vector.h:552)
                                    ▼ 49.9 MiB: WireCell::Aux::SimpleTrace::SimpleTrace(int, std::vector<> const&) (SimpleTrace.cxx:8)
                                      ▼ 49.8 MiB: aligned_malloc (Memory.h:182)
                                      ▼ 41.1 MiB: allocate (new_allocator.h:114)
                                        ▼ 41.1 MiB: allocate (alloc_traits.h:444)
                                          ▼ 41.1 MiB: M_allocate (stl_vector.h:343)
                                            ▼ 41.1 MiB: M_allocate_and_copy<> (stl_vector.h:1508)
                                              ▼ 41.1 MiB: std::vector<>::operator=(std::vector<> const&) (vector.tcc:226)
                                                ▼ 34.1 MiB: WireCell::Gen::EmpiricalNoiseModel::channel_spectrum(int) const (EmpiricalNoiseModel.cxx:436)
                                                  ▼ 34.1 MiB: WireCell::Gen::incoherentAddNoise::operator()(std::shared_ptr<> const&, std::shared_ptr<> s) (AddNoise.cxx:80)
                                                    ▼ 7.0 MiB: WireCell::Gen::EmpiricalNoiseModel::resample(WireCell::Gen::EmpiricalNoiseModel::NoiseSpectrum&) const (EmpiricalNoiseModel.cxx:276)
                                                      ▼ 37.9 MiB: aligned_malloc (Memory.h:182)
                                                      ▼ 33.4 MiB: allocate (new_allocator.h:114)
                                                        ▼ 33.4 MiB: allocate (alloc_traits.h:444)
                                                          ▼ 33.4 MiB: M_allocate (stl_vector.h:343)
                                                            ▼ 33.4 MiB: M_create_storage (stl_vector.h:358)
                                                              ▼ 33.4 MiB: _Vector_base (stl_vector.h:302)
                                                                ▼ 33.4 MiB: vector (stl_vector.h:552)
                                                                  ▼ 33.4 MiB: pair<> (stl_pair.h:342)
                                                                    ▼ 33.4 MiB: WireCell::Gen::EmpiricalNoiseModel::interpolate_wire_length(int, double) const (EmpiricalNoiseModel.cxx:276)
                                                                      ▼ 33.4 MiB: WireCell::Gen::EmpiricalNoiseModel::get_spectrum_data(int, int) const (EmpiricalNoiseModel.cxx:319)
                                                                        ▼ 19.5 MiB: clang::ModuleMap::findOrCreateModule(Illvm::StringRef, clang::Module*, bool, bool) (in /cvfms/larsoft.opensciencgrid.org/products/root/v6_22_06...
                                                                          ▼ 18.9 MiB: aligned_malloc (Memory.h:182)
                                                                          ▼ 18.9 MiB: aligned_malloc (Memory.h:182)
                                                                          ▼ 13.3 MiB: allocate (new_allocator.h:114)
                                                                          ▼ 13.0 MiB: allocate (new_allocator.h:114)
                                                                          ▼ 6.1 MiB: llvms::SmallVectorBase::grow_pod(void*, unsigned long, unsigned long) (in /cvfms/larsoft.opensciencgrid.org/products/root/v6_22_06a/Linux64bit...
                                                                      ▼ 186.5 MiB: Snapshot #43
                                                                      ▼ 532.6 MiB: Snapshot #55
                                                                      ▼ 551.5 MiB: Snapshot #56 (peak)

```

Custom Allocators   Massif Data

```

30 ... // fixme: see https://github.com/WireCell/wire-cell-gen/issues/29
31 ... local make_noise_model = function(anode, csdb=null) {
32 ...     type: "EmpiricalNoiseModel",
33 ...     name: "emperiricalnoise%s"% anode.name,
34 ...     data: {
35 ...         anode: wc.tn(anode),
36 ...         dft: wc.tn(tools.dft),
37 ...         chanstat: if std.type(csdb) == "null" then "" else wc.tn(csdb),
38 ...         spectra_file: params.files.noise,
39 ...         nsamples: params.daq.nticks,
40 ...         period: params.daq.tick,
41 ...         wire_length_scale: 1.0*wc.cm, // optimization binning
42 ...     },
43 ...     uses: [anode, tools.dft] + if std.type(csdb) == "null" then [] else [csdb],
44 ... },
45 ... local noise_models = [make_noise_model(anode) for anode in tools.anodes],
46
47
48 ... local add_noise = function(model) g.pnode({
49 ...     type: "AddNoise",
50 ...     name: "addnoise%s"%[model.name],
51 ...     data: {
52 ...         rng: wc.tn(tools.random),
53 ...         dft: wc.tn(tools.dft),
54 ...         model: wc.tn(model),
55 ...         nsamples: params.daq.nticks,
56 ...         replacement_percentage: 0.02, // random optimization
57 ...     }}, nin=1, nout=1, uses=[tools.random, tools.dft, model]),
58
59 ... local noises = [add_noise(model) for model in noise_models],

```

Each APA cache its own noise spec.