

Hello everybody,

Yesterday two problems were discovered. Both are related to a change in the prespec package, that I did on 10th of October 2014. I'll try to explain what happened. I do it here, because the problem is related to the spikes in Ricardo's spectrum (see his initial post in this topic).

I silently introduced two problematic changes in the AgataAdapter (prespec/plugins/AGATA/process/AgataAdapter.cpp) and the Psa unpacker (prespec/plugins/AGATA/unpack/Psa.cpp) into the prespec repository:

1. For a reason that I don't remember, I changed the meaning of the parameter "psa\_hit\_t\_source" in the AgataAdapter. This parameter decides which time value is copied into the time value of psa\_hit\_t. Before the change, there were 3 options:
  - a) `psa_hit_t_source == 0`: `psa_hit_t` is coming from the values measured by multihit-TDCs in the AGATA crate
  - b) `psa_hit_t_source == 1`: `psa_hit_t` is computed from the T1 in the psa crystal frame (the time from low-gain core signal), the GTS timestamp difference, and the time difference between Sc41 and AGAVA accepted trigger
  - c) `psa_hit_t_source == 2`: `psa_hit_t` is computed from the T0 in the psa crystal frame (the time from high-gain core signal), the GTS timestamp difference, and the time difference between Sc41 and AGAVA accepted trigger (The difference between b) and c) is only T1 and T0)

After the change there were four options

- a) `psa_hit_t_source == 0`: `psa_hit_t` is coming from the time values in the psa frame
- b) `psa_hit_t_source == 1`: same as `psa_hit_t_source == 0` (before the change)
- c) `psa_hit_t_source == 2`: same as `psa_hit_t_source == 1` (before the change)
- d) `psa_hit_t_source == 3`: same as `psa_hit_t_source == 2` (before the change)

This affected the behavior of the analysis after pulling these changes. I didn't mention this in the forum or even in the git commit message. This is really bad, because by this I have changed the behavior of any analysis that uses the AgataAdapter (the one inside the AGATA plugin). I have broken the rule of not introducing changes to behavior of existing processors.

2. The other change was introducing a subtle, but even more severe problem. The AgataAdapter gets input only from the Psa unpacker. Before the code change, the time information in the psa frame was ignored by the Psa unpacker, and also ignored in the AgataAdapter. In order to examine the time information of psa frames I decided to change both, the Psa unpacker and the AgataAdapter inside the AGATA plugin to handle the time values in the psa frames (and by this introduced the need for a 4th option for the parameter `psa_hit_t_source`). I was assuming that this change did not affect anybody, because the Psa unpacker and the AgataAdapter are always used together.

However, I did not take into account, that there are older versions of Psa unpacker and AgataAdapter in use (especially in the configuration files from 2012 campaign). If your configuration was using an old Psa unpacker and the modified AgataAdapter, the result is that

all but the first psa hit data is corrupted (the spectra have spikes). In the case of MGT or any other algorithm that makes use of the psa hit position and energy information this will completely screw up the result.

Your analysis was affected in the following cases:

- \* using old Psa unpacker with new AgataAdapter.
- \* using new Psa unpacker with old AgataAdapter.

Your analysis was not affected in the following cases:

- \* using old Psa unpacker with old AgataAdapter.
- \* using new Psa unpacker with new AgataAdapter.

By "new" I mean any unpacker or processor, that inside one of the plugins AGATA/FRS/LYCCA/....

You can identify the "new" unpackers or processors in the config file by the syntax "PLUGINNAME.unpackername" or "PLUGINNAME.processorname"

```
crate anyCrate
  module Psa # an old unpacker
end
```

```
using AGATA libprespecAGATA.so
crate anyCrate
  module AGATA.Psa # a new unpacker
end
```

Finally, I can only say that I really regret doing these changes... but I cannot revert time, so all I can do is apologize for any inconvenience that these have induced.

Michael