Progress with PRs – merged

- PR #801 (AV, based on Jorgen's PR #774) – MERGED (approved OM)
  - Add GpuAbstraction.h (including HIP implementation) and HIP makefile targets
  - Tested on AMD GPUs on the LUMI HPC

- PR #811 (AV, based on Olivier's mg5amcnlo changes) – MERGED (approved OM)
  - Update mg5amcnlo submodule to Olivier's most recent gpucpp branch commits

- PR #368 (AV) – MERGED (approved OM)
  - Remove gXXX.cu symlinks for XXX.cc files in generated code
  - Build XXX_cu.o from XXX.cc using "nvcc –x cu"
  - Build XXX_cu.o from XXX.cc using "hipcc –x hip"
    - Eventually: XXX_hip.o instead? I think it would be better...
    - (And eventually split also XXX_cppnone.o, XXX_cppavx2.o etc? I think probably not...?)

Progress with PRs – ready to merge

- PR [#809](#) (AV) – ~READY TO MERGE (approved OM, waiting for SR approval)
  - Add support for ROCRAND on AMD GPU (as suggested by SR)
  - Implemented via the HIPRAND API (drop-in replacement for CURAND)
    - Implemented: random numbers on GPU device
    - Not yet implemented (WIP in HIPRAND): random number on CPU host, random number ordering

- PR [#796](#) (AV) – ~READY TO MERGE (finalizing discussions with SR)
  - Add fbridgesequence_nomultichannel function, needed by SR in channel array PR
  - Can be merged independently of the channel array PR (and I suggest we do)

Progress with PRs – almost ready or WIP

- PR #813 (OM) – ~ALMOST READY TO MERGE (waiting for AV review)
  - OM: moved handling of a few files (counters.cc etc) from patchMad.sh to 'proper' plugin
  - AV: this was ok for .MAD but added unnecessary files in .SA, added an ugly hack to fix that
  - OM: implemented a better fix than AV's ugly hack... waiting for review

- PR #798 (AV, based on Jorgen's #775) – ~ALMOST READY (must fix new conflicts)
  - Separate build targets for CUDA and C++ (and must now add HIP)
  - This was complete and ready to merge before recent merges
    - Now there are a few ~easy conflicts to fix (HIP, HIPRAND, gXXX.cu all changed makefiles)
  - As discussed last week: suggest to merge this before selected changes in SH's PR #753

- Issue #765 (SR's new_interface_wrap branch, no PR yet) – WIP?
  - From scalar channel ID to array of channel IDs
  - Will use AV's fbridgesequence_multichannel in PR #796
  - Eventually need also Olivier's mg5amcnlo gpucpp_wrap (not yet in gpucpp): complete?

madgraph4gpu vs. mg5amcnlo_cudacpp

- The PAST and PRESENT: https://github.com/madgraph5/madgraph4gpu (up to date)
  - This contains the code generation plugin but also tests and generated code
  - PRs currently still all go here
  - Our issue tracker is still here
  - The old CI (SH) uses generated code, the new CI (AV, to be extended) uses the plugin only
  - OM: it is difficult to review PRs here because also tests and generated code change

- The FUTURE: https://github.com/mg5amcnlo/mg5amcnlo_cudacpp (6 months old)
  - This contains ONLY the code generation plugin

- OM suggests: remove tests and generated code from the old repo to ease PR reviews

- AV suggests instead: let's move to the new repo asap (let's drain existing PRs...)
  - Personally I definitely want to keep an old repo as is with tests and generated code
  - Todo: update new from old repo (or better: I can recreate it from scratch if you prefer?)
    - Todo: scripts to update new from old, keeping history (I have ~scripts for this, uncommitted)
    - Todo: scripts to update old from new, keeping history (if we merge directly in the new repo)
  - Question: when do we consider the new repo the reference and the old repo a legacy?
  - Todo: add AV's "new" CI tests to new repo (main issue, set up GPU nodes – SH?)
  - Also todo: add cudacpp module under PLUGIN in https://github.com/mg5amcnlo/mg5amcnlo
    - Todo? Implement "minimal", "maximal", "last validated" checks based on github commits???

Backup – from last week

Makefiles – two options forward?

- Option 1 (one-step) – start from Stephan's PR #753 (itself based on Jorgen's #775)
  - Advantages:
    - Stephan's commits are immediately included
  - Disadvantages:
    - One-step approach mixes logically different features – debugging issues is more complex
    - Inherits issues to be fixed in Jorgen's PR – disentangling and debugging issues is more complex
    - No HIP support until all makefiles issues are solved – will take longer, Nathan is waiting for it

- Option 2 (two-steps) – start from Jorgen's #775 first, later add Stephan's PR #753
  - Advantages:
    - Debug issues in Jorgen's PR separately from any issues in Stephan's commits, logically cleaner
    - HIP support may be added before all issues in Stephan's PR are fixed, Nathan may work earlier
  - Disadvantages:
    - Must fix conflicts in the merge, or (better) add back Stephan's features one by one

- I would most certainly go for option 2 – technically more appropriate

Other missing items before a release

- Update the plugin (issue [#661](#)) – or recreate it with the full history?
  - [mg5amcnlo_cudacpp](#) exists with the full history but is stuck to Aug 30

- Do we want to try and fix some EFT and SUSY before the release?

- See also the May 2023 summary (issue [#671](#)): not up to date but still relevant
  - The issues that are still open remain desirable, though not strictly necessary
  - Many issues mentioned there have been fixed/completed
  - Only a few new issues have appeared (e.g. channel id array)

- Am I missing some very big thing not mentioned above?