Status of PRs towards a release

(plus CMS DY+jets and a few other things)

Andrea Valassi (CERN)

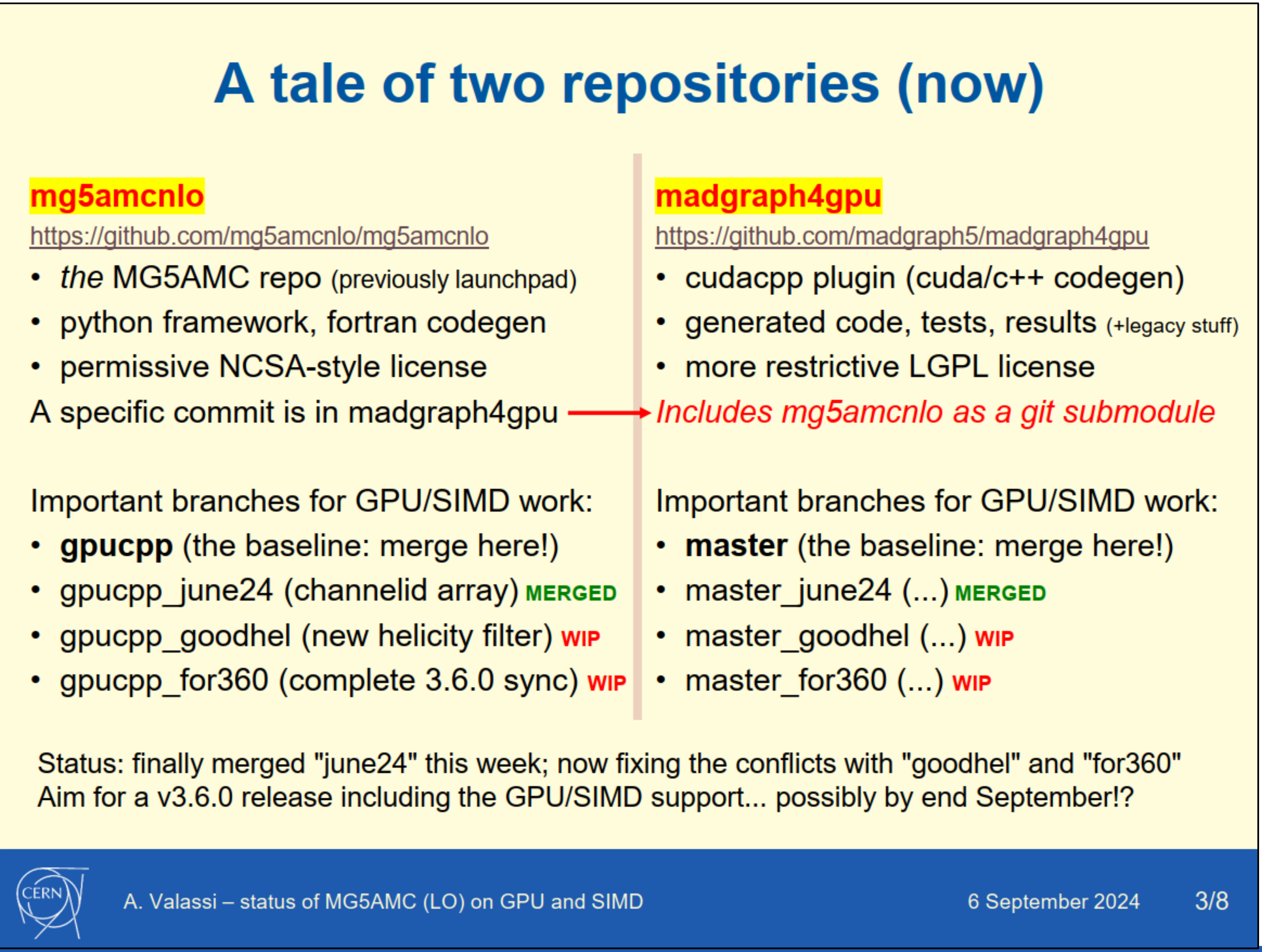
(describing also work done with and/or by Olivier - thanks!)

Madgraph on GPU development meeting, 17th September 2024

https://indico.cern.ch/event/1355162

(previous update was last week on September 17 – only mentioning changes since then)

Packaging – 4 weeks ago



A tale of two repositories (now)

mg5amcnlo

https://github.com/mg5amcnlo/mg5amcnlo

- *the* MG5AMC repo (previously launchpad)
- python framework, fortran codegen
- permissive NCSA-style license

A specific commit is in madgraph4gpu →

Important branches for GPU/SIMD work:

- **gpucpp** (the baseline: merge here!)
- gpucpp_june24 (channelid array) MERGED
- gpucpp_goodhel (new helicity filter) WIP
- gpucpp_for360 (complete 3.6.0 sync) WIP

madgraph4gpu

https://github.com/madgraph5/madgraph4gpu

- cudacpp plugin (cuda/c++ codegen)
- generated code, tests, results (+legacy stuff)
- more restrictive LGPL license

Includes mg5amcnlo as a git submodule

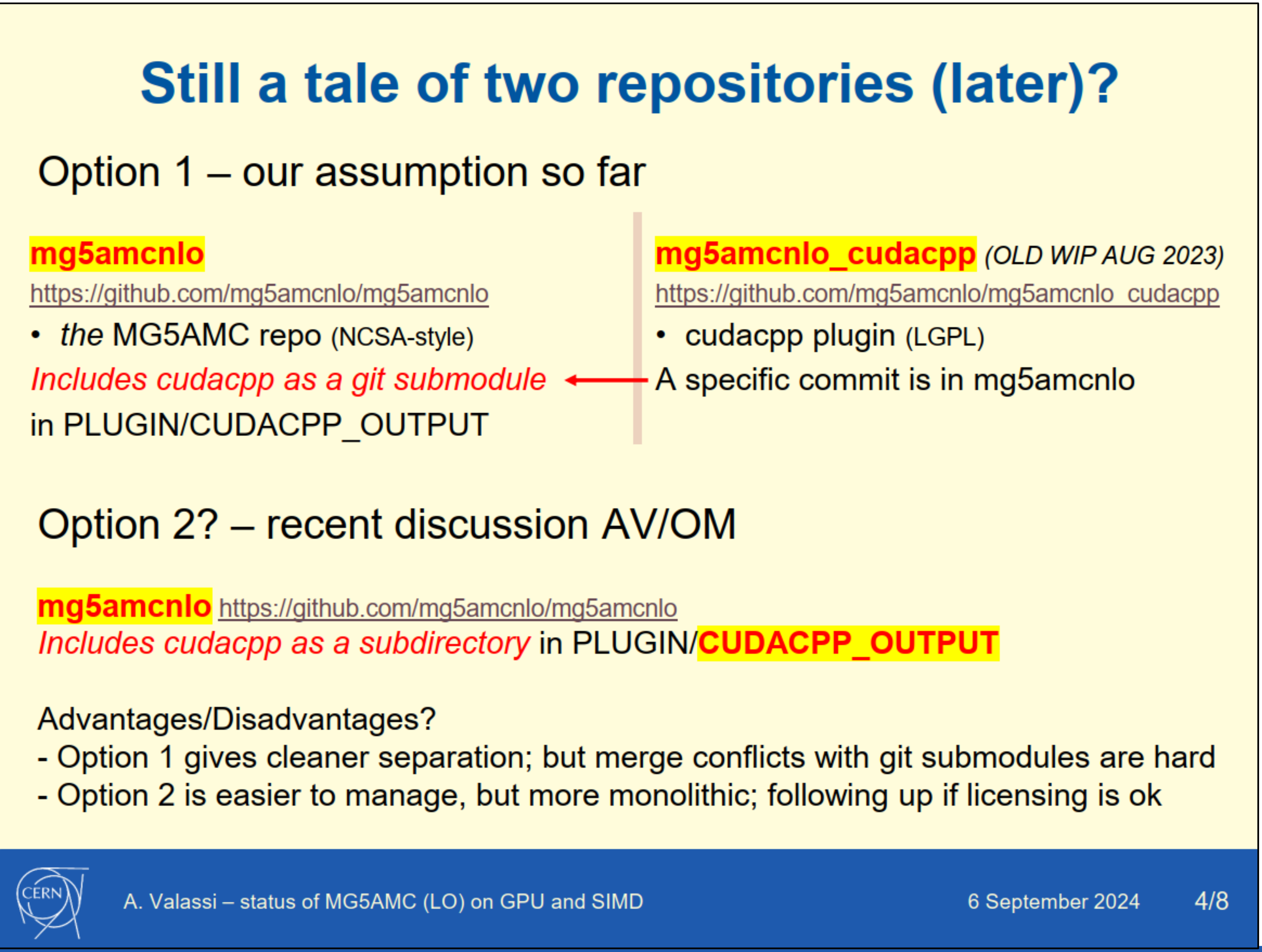
Important branches for GPU/SIMD work:

- **master** (the baseline: merge here!)
- master_june24 (...) MERGED
- master_goodhel (...) WIP
- master_for360 (...) WIP

Status: finally merged "june24" this week; now fixing the conflicts with "goodhel" and "for360"
Aim for a v3.6.0 release including the GPU/SIMD support... possibly by end September!?

A. Valassi – status of MG5AMC (LO) on GPU and SIMD 6 September 2024 3/8

Packaging – 4 weeks ago



Still a tale of two repositories (later)?

Option 1 – our assumption so far

mg5amcnlo
https://github.com/mg5amcnlo/mg5amcnlo
- *the* MG5AMC repo (NCSA-style)

Includes cudacpp as a git submodule ← A specific commit is in mg5amcnlo
in PLUGIN/CUDACPP_OUTPUT

mg5amcnlo_cudacpp *(OLD WIP AUG 2023)*
https://github.com/mg5amcnlo/mg5amcnlo_cudacpp
- cudacpp plugin (LGPL)

A specific commit is in mg5amcnlo

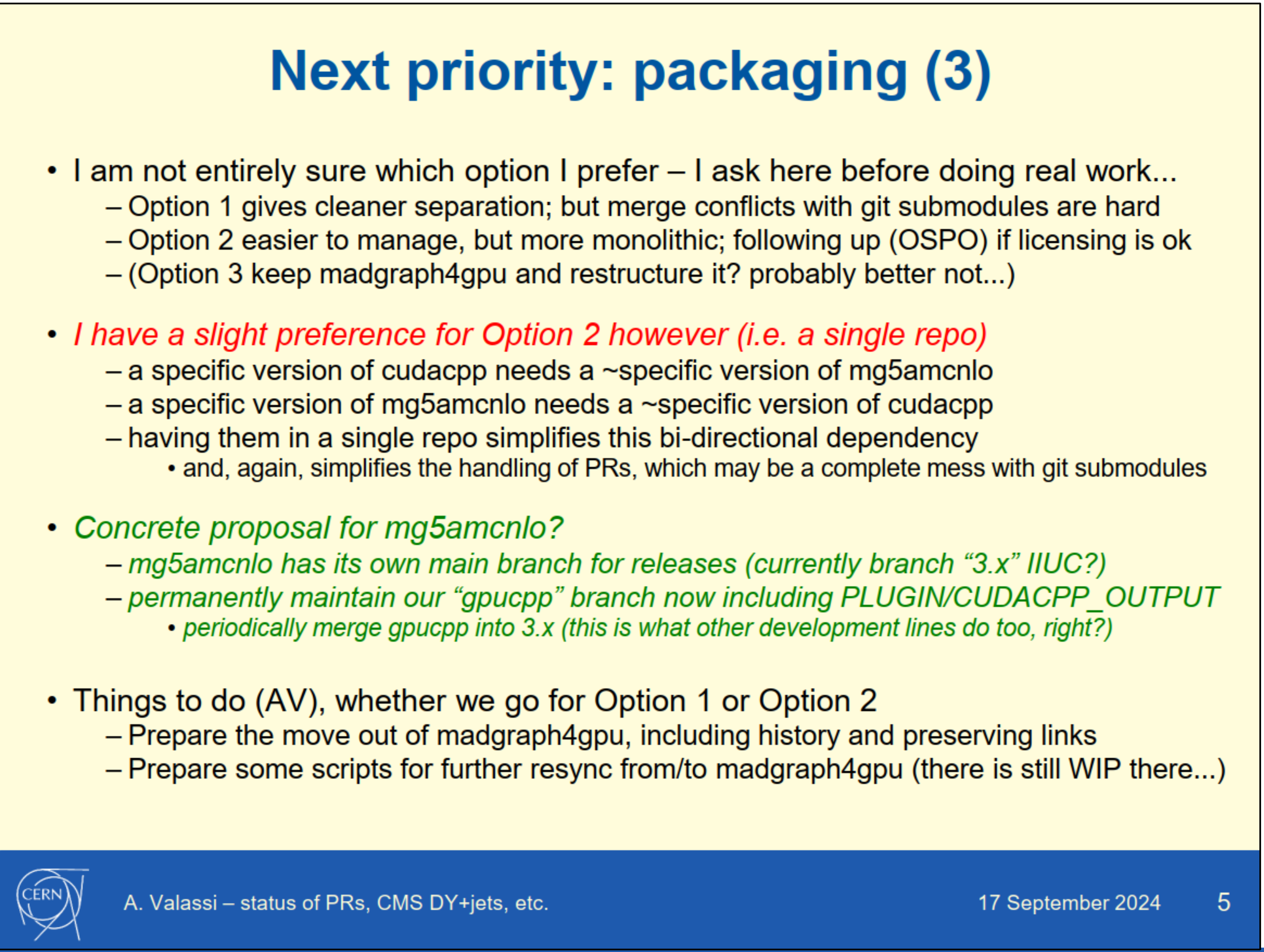
Option 2? – recent discussion AV/OM

mg5amcnlo https://github.com/mg5amcnlo/mg5amcnlo
Includes cudacpp as a subdirectory in PLUGIN/**CUDACPP_OUTPUT**

Advantages/Disadvantages?
- Option 1 gives cleaner separation; but merge conflicts with git submodules are hard
- Option 2 is easier to manage, but more monolithic; following up if licensing is ok

CERN A. Valassi – status of MG5AMC (LO) on GPU and SIMD 6 September 2024 4/8

Packaging – two weeks ago



Next priority: packaging (3)

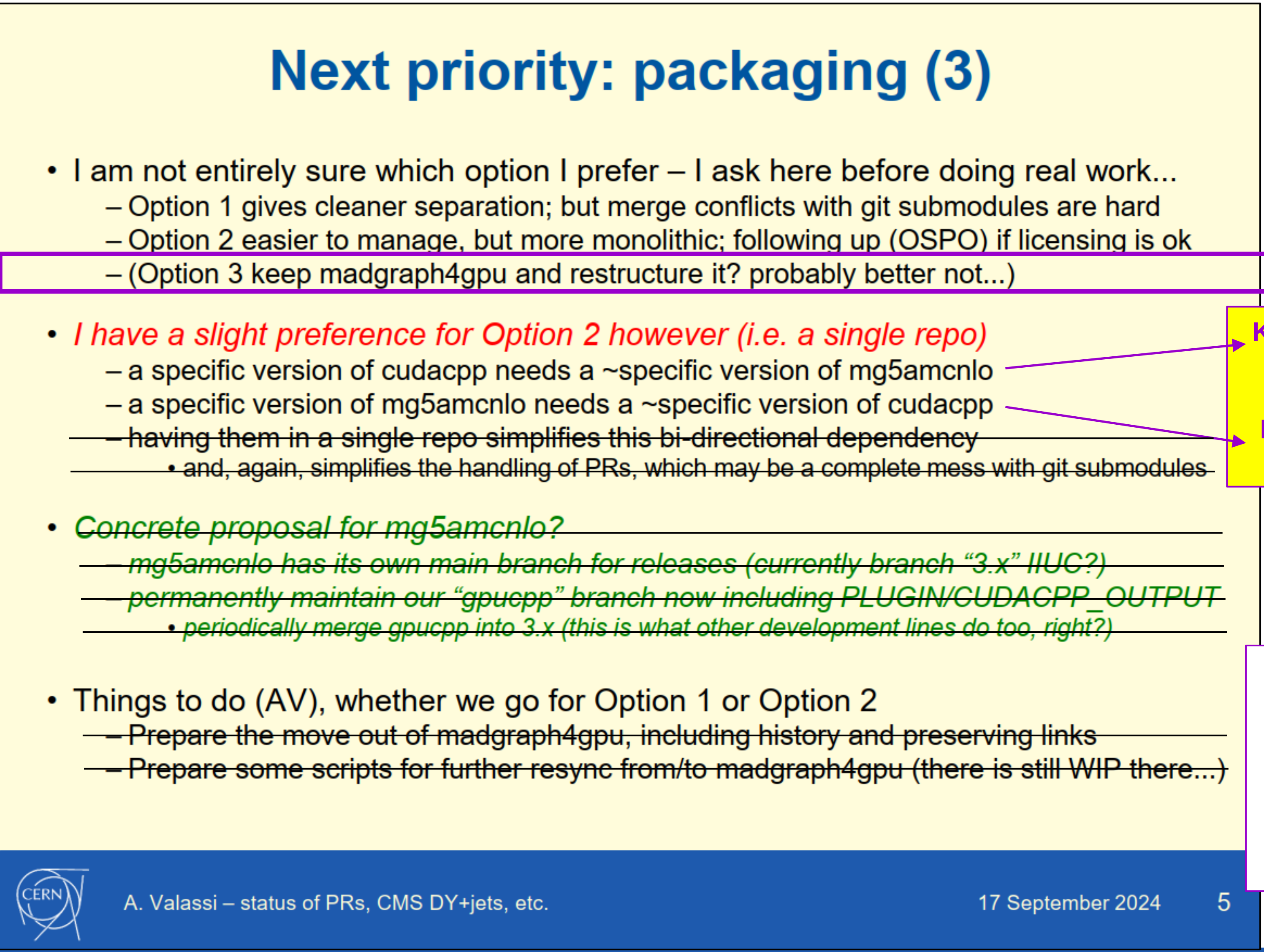
- I am not entirely sure which option I prefer – I ask here before doing real work...
  - Option 1 gives cleaner separation; but merge conflicts with git submodules are hard
  - Option 2 easier to manage, but more monolithic; following up (OSPO) if licensing is ok
  - (Option 3 keep madgraph4gpu and restructure it? probably better not...)
- *I have a slight preference for Option 2 however (i.e. a single repo)*
  - a specific version of cudacpp needs a ~specific version of mg5amcnlo
  - a specific version of mg5amcnlo needs a ~specific version of cudacpp
  - having them in a single repo simplifies this bi-directional dependency
    - and, again, simplifies the handling of PRs, which may be a complete mess with git submodules
- *Concrete proposal for mg5amcnlo?*
  - *mg5amcnlo has its own main branch for releases (currently branch “3.x” IIUC?)*
  - *permanently maintain our “gpucpp” branch now including PLUGIN/CUDACPP_OUTPUT*
    - *periodically merge gpucpp into 3.x (this is what other development lines do too, right?)*
- Things to do (AV), whether we go for Option 1 or Option 2
  - Prepare the move out of madgraph4gpu, including history and preserving links
  - Prepare some scripts for further resync from/to madgraph4gpu (there is still WIP there...)

CERN A. Valassi – status of PRs, CMS DY+jets, etc. 17 September 2024 5

Packaging – ~~two weeks ago~~ *now*

Very useful discussion at the meeting 17 Sep (and later with OM)

And the answer is... ~ Option 3!



Next priority: packaging (3)

- I am not entirely sure which option I prefer – I ask here before doing real work...
  - Option 1 gives cleaner separation; but merge conflicts with git submodules are hard
  - Option 2 easier to manage, but more monolithic; following up (OSPO) if licensing is ok
  - (Option 3 keep madgraph4gpu and restructure it? probably better not...)
- *I have a slight preference for Option 2 however (i.e. a single repo)*
  - a specific version of cudacpp needs a ~specific version of mg5amcnlo
  - a specific version of mg5amcnlo needs a ~specific version of cudacpp
  - ~~having them in a single repo simplifies this bi-directional dependency~~
    - ~~and, again, simplifies the handling of PRs, which may be a complete mess with git submodules~~
- ~~*Concrete proposal for mg5amcnlo?*~~
  - ~~*mg5amcnlo has its own main branch for releases (currently branch “3.x” IIUC?)*~~
  - ~~*permanently maintain our “gpucpp” branch now including PLUGIN/CUDACPP_OUTPUT*~~
    - ~~*periodically merge gpucpp into 3.x (this is what other development lines do too, right?)*~~
- Things to do (AV), whether we go for Option 1 or Option 2
  - ~~Prepare the move out of madgraph4gpu, including history and preserving links~~
  - ~~Prepare some scripts for further resync from/to madgraph4gpu (there is still WIP there...)~~

A. Valassi – status of PRs, CMS DY+jets, etc. 17 September 2024 5

Keep mg5amcnlo as a git submodule in cudacpp

Download cudacpp as a tarball into mg5amcnlo

May keep madgraph4gpu ~as-is for the moment (and clean it later)

Issues and PRs remain

Commit history remains

A tale of two repositories (again)

TODO on madgraph4gpu (some of this may be done after the release!)
Remove legacy stuff (e.g. epoch0, epoch1; alpaka, kokkos; sycl??...)
Simplify the directory structure
Rename the repo (from madgraph4gpu to mg5amcnlo_gpucpp?)
Move the repo (from madgraph5 to mg5amcnlo)
...

mg5amcnlo

https://github.com/mg5amcnlo/mg5amcnlo

- *the* MG5AMC repo (previously launchpad)
- python framework, fortran codegen
- permissive NCSA-style license

A specific commit is in madgraph4gpu →

Important branches for GPU/SIMD work:

- **3.x** (the release: now includes gpucpp!)
- **gpucpp** (still the baseline, merge here?)

~~madgraph4gpu~~

RENAME! MOVE!

~~https://github.com/madgraph5/madgraph4gpu~~

- cudacpp plugin (cuda/c++ codegen)
- keep generated code, tests, results?
- keep sycl in a separate directory?
- more restrictive LGPL license

Includes mg5amcnlo as a git submodule

Important branches for GPU/SIMD work:

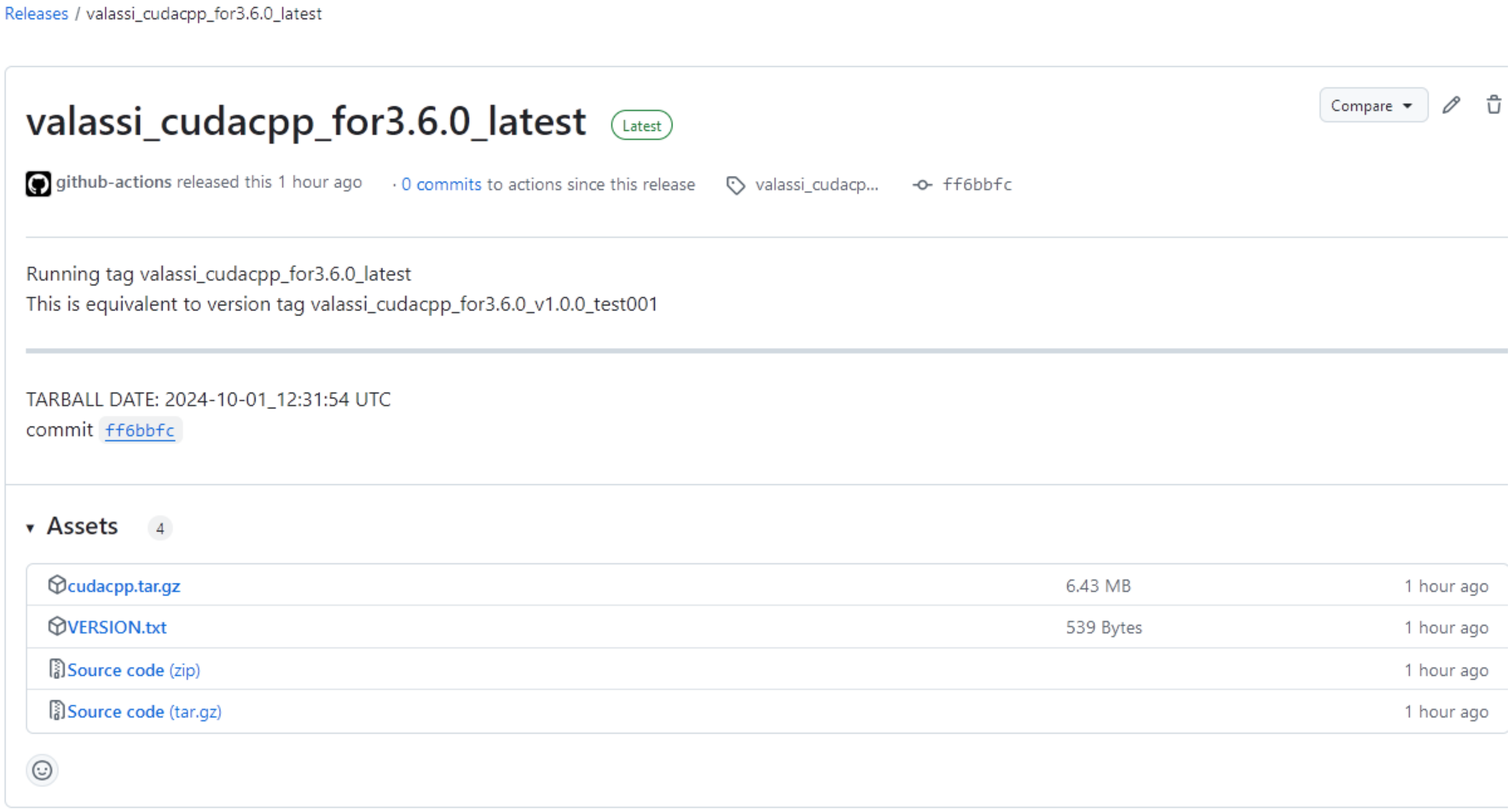
- **master** (the baseline: merge here!)

Quick overview of recent work (AV + OM)

- Will give details in PR #1008
- Manual script: “./gitTag.sh v1.0.0_pre1”
  - Creates tag “cudacpp_for3.6.0_v1.0.0_pre1” (finds 3.6.0, checks 1.0.0)
- CI machinery “archiver”
  - Triggered on new tags “cudacpp_for*”
  - Creates cudacpp.tar.gz and VERSION.txt
  - Creates github release from git tag, uploads cudacpp.tar.gz and VERSION.txt
  - Creates a second running git tag “cudacpp_for3.6.0_latest”
  - Creates a second github release from this second git tag, uploads the same files
  - Creates version_info.dat
    - This links 3.6.0 to cudacpp.tar.gz for “cudacpp_for3.6.0_latest
- HEPToolsInstaller patches
  - Initially hardcoded rules to find cudacpp.tar.gz for “cudacpp_for3.6.0_latest
  - Now back to Olivier’s preferred mechanism using version_info.dat

Example release

https://github.com/valassi/madgraph4gpu/releases/tag/valassi_cudacpp_for3.6.0_latest



Releases / valassi_cudacpp_for3.6.0_latest

valassi_cudacpp_for3.6.0_latest

github-actions released this 1 hour ago · 0 commits to actions since this release valassi_cudacp... ff6bbfc

Running tag valassi_cudacpp_for3.6.0_latest
This is equivalent to version tag valassi_cudacpp_for3.6.0_v1.0.0_test001

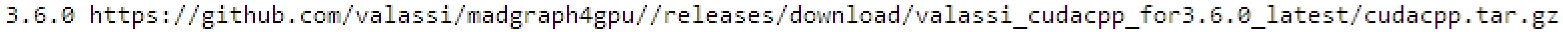
TARBALL DATE: 2024-10-01_12:31:54 UTC
commit ff6bbfc

▾ Assets 4

cudacpp.tar.gz	6.43 MB	1 hour ago
VERSION.txt	539 Bytes	1 hour ago
Source code (zip)		1 hour ago
Source code (tar.gz)		1 hour ago

Example version_info.dat

https://raw.githubusercontent.com/valassi/madgraph4gpu/refs/heads/INFO/version_info.dat



```
3.6.0 https://github.com/valassi/madgraph4gpu//releases/download/valassi_cudacpp_for3.6.0_latest/cudacpp.tar.gz
```