Update on Software Licenses

Benedikt Hegner, Graeme Stewart
(CERN)

HSF Startup Team Meeting
14.9.2017

- We ignored the topic of copyright and licenses for too long
- In the current situation our experiment stacks are in not well-defined state:
  - Generators using GPLv2, which is not compatible with other popular licenses - notably (L)GPLv3 and Apache 2
  - E.g., as it stands ATLAS cannot open source, or even release, Athena as there is no compatible license with all the externals used
- In total our community has O(200) projects with O(10) licenses

⇒ Need to make sure that we have a consistent and compatible license situation within the community

- In passing, the issue of licensing has nothing at all to do with attribution (in fact *insisting* on attribution violates most open source licenses)
- We strongly encourage software attribution and citation

- Software is owned by its copyright holders
  - They can grant others rights for using their software (licenses)
- Licenses can only be decided with the agreement of all copyright holders
  - No license possible without clear copyright situation
  - Usually not well defined for HEP projects
  - Problems in our field:
    - High number of copyright owners (individuals, institutes, …)
    - Plenty of copyright owners not in the field any more
  - Contribution guides can clarify that contributions transfer copyright to help with this
- Only legal entities can hold copyright, which excludes experiment collaborations and the HSF
- If copyright can be assigned to one single organisation then the practicalities of licensing become far easier
  - This is the approach adopted by ATLAS and CMS
  - CERN holds copyright ‘for the benefit’ of the experiments, to permit the widest possible adoption and reuse

Compatibility of license X, when using software under license Y

X	Y: Public Domain	MIT/X11/Boost	BSD	Apache 2	Mozilla 2	LGPL 2.1	LGPL 3	GPLv2	GPLv3
Public Domain	No change	No change	No change	No change	No change	No change	No change	**Change**	**Change**
MIT/X11/Boost	No change	No change	No change	No change	No change	No change	No change	**Change**	**Change**
BSD	No change	No change	No change	No change	No change	No change	No change	**Change**	**Change**
Apache 2	No change	No change	No change	No change	No change	Not possible**	No change	Not possible**	**Change**
Mozilla 2	No change	No change	No change	No change	No change	No change	No change	**Change***	**Change***
LGPL 2.1	No change	No change	No change	Not possible**	No change	No change	No change	**Change**	Not possible**
LGPL 3	No change	No change	No change	No change	No change	No change	No change	Not possible**	**Change**
GPLv2	No change	No change	No change	Not possible**	No change*	No change	Not possible**	No change	Not possible**
GPLv3	No change	No change	No change	No change	No change*	No change	No change	Not possible**	No change

Note that the GPLv2+ license can be taken as GPLv2 or GPLv3, but not both

* In this special case the code must be dual licensed and the MPL2 code must allow for that

** Incompatible due to patent handling clauses

- See our license compatibility spreadsheet and the (L)GPL compatibility matrix
- Note that the GPLv2 and GPLv3 licenses are the ‘viral’ ones, meaning they change the license of their clients
- For the experiment software, these are large ‘derivative works’ so have to be released under a license compatible with all their dependencies

To decide on a license one has to consider two questions

1. What constraints do I want to put onto users of my package?
2. What constraints do externals I use already put onto me?

1. Constraints to be put onto users*

- Do improvements to the code itself need to be made public? (GPL, LGPL, MPL, ...)
- Does software that uses my package need to be open-source as well? (GPL)
- Do I want to leave users the choice to pick a license themselves? (MIT, Apache LGPL, MPL, ...)

2. Constraints imposed by external software

- Is my license compatible with the license of SW I am using?
- Do I rely on GPL software ? → your software is GPL as well

*Private changes are always allowed, but we don't want to encourage code to be private

- Theory Community
  - Currently using GPLv2; proposing to moving to GPLv2+ or GPLv3
- HepMC
  - Updated HepMC 2 license from GPLv2 to GPLv2+
  - Updated HepMC 3 license from GPLv3 to GPLv3+
  - We think this is a good model, if the GPL has been chosen
- FastJet
  - Investigating licensing under GPLv2 and GPLv3
    - They have a problem that some of their plugins are GPLv2 and some are GPLv3
- ACTS
  - Using Mozilla Public License v2 (aka MPL)
  - We had some concerns about incompatibility with dual licensing, needed for combining with GPL licenses
    - However, after a careful reading of the license these were allayed

The HSF project template provides GPLv3, LGPLv3 and Apache 2.0 as the three options.

- ATLAS
  - Exploring going for GPLv3 for SW stack
  - Choice driven by external software
- LHCb:
  - Approach similar to ATLAS
- CMS:
  - Considering mixed licensing:
    - GPLv3 for software depending on GPL externals
    - Apache 2.0 for subset not depending on these externals
- FCC:
  - GPLv3
- Belle-2
  - Formed a license working group

HSF license working group released a technical note in Feb 2016, describing the various OSS licenses.

- Find a consensus between LHC experiments (at least) on how we approach licensing our experiment software stacks
  - Does mixed licensing work?
- Follow up with OpenLab how much of a problem GPL would represent to OpenLab partners
- Engage with the wider theory community about adopting GPLv3 more widely
  - Directly or as GPLv2+
  - Use contacts via GENSER project
- Encourage better software citation/ train people on best practices
- Update the HSF technical note with our knowledge of some of the practical considerations that we now understand better than before