

# Introduction to discussion

*Internal Seminars & Generation Transfer Knowledge*

27<sup>th</sup> Geant4 Collaboration Meeting,

26<sup>th</sup> - 30<sup>th</sup> September 2022,

Rennes,

Marc Verderi (LLR), Lorenzo Pezzotti (CERN)

# This session



- An introductory presentation
  - This one
- An open discussion
- The first official “internal seminar”
  - *“An Introduction to GPUs and their Applicability to MC”*
  - By Jonas
    - Many thanks to him !
  - Hoping this “first” seminar will indeed be followed by many others

# How this started



- During the program committee preparatory meetings (members : Ben Morgan, Jonas Hahnfeld, Anna Zaborowska, Lorenzo Pezzotti, Pedro Arce + ex-officio), the point of ***“internal seminars”*** came
- Came as a way to
  - Share knowledge
  - Foster emergence of new ideas
  - Improve collaboration spirit
  - **Help with transfer knowledge from “senior” to “junior” generation**
    - Transfer knowledge on the physics models
    - On their implementation
    - Without limiting to physics, but extending to kernel codes
    - Ie transfer knowledge to people who will continue developing and maintaining these codes
    - **This demand was strongly expressed**
      - And “**please read the documentation & please read the code**” is not an adequate answer ;)
- And ***“generation transfer knowledge”*** became an item by itself
- We decided to bring these discussion points at the Collaboration level

# Generation Transfer Knowledge



- Almost all working groups need to **embark junior developers to take over**
  - And an **overlap period long enough is needed** to perform “**generation transfer knowledge**” !
    - If such long overlap is not possible, then it is easy : **game over !**
  - **But let's assume that this malediction (game over) is not 100% granted and let's even dream that addressing the transfer knowledge issue may help to exorcise that malediction.**
- On a software developed during  $O(30)$  years many (many) things can be said
- **“Senior generation” people have to be aware that many features that sound “usual” for them –because they developed these– are “new” to the “junior generation” !**
  - And many of/almost all the considerations & discussion made to guide the choices to come to these features are unknown to the “junior generation” !
- **All this represents a lot of experience and thoughts:**
  - Why such model has been chosen and not such competitor ?
  - Were the choices made successful or disappointing by some respects ?
    - Were even some attempts retracted and for what reasons ?
  - If alternative(s) would have to be considered which one(s) would be worth investigated ?
  - Etc.
  - **Such considerations are rarely documented –and are uneasy to document – but represent a considerable amount of experience accumulated over time.**

# Generation Transfer Knowledge



- If the previous is very true for the physics, **it applies to all domains of Geant4**
  - Why did we chose the modeling we have for the geometry ?
    - What strengths and what weaknesses ?
  - What made us to decide to come to the tracking as it is ?
    - What do we like in the approach adopted and what would we dream to change or revolutionize ?
  - What do we like in the design of the processes (G4VProcess) ?
    - And what do we think we missed ?
      - **For example EM and Hadronic both invented “models” but independently, with duplication or similar/same functionalities and no genericity between the two implementations**
      - We realized over time that modeling of physics need to be flexible –per energy range, per region,...-
      - We realized over time that we would need flexible treatment of physics with “analog” treatment in some parts and “biased” one in some others
      - We realized over time that it would be nice to be able to happily mix all those in applications, easily, flexibly
      - Etc.
  - But also all infrastructure of testing tools, software management, software validation, physics validation
    - **We do not have a clear “collaboration wide physics validation suite” in operation today !**
    - **There is a lot (lot) of experience too in these “technical” supports provided for developments and validations !**
- All these considerations may sound like a “legacy” –**they are**– but are also a “potting soil” for **fostering emergence of new ideas, based on educated considerations**, not reinventing the wheel
  - **A simple question of efficacy ! Not of interest only for “junior generation” but to “senior one” too !**
  - Eg : Dmitri and Lorenzo received quite support requests on geant-val from “senior” this week !
- **(quite easy question) Do we share the above considerations ?**
- **(more uneasy question) How do we address this “Generation Transfer Knowledge” ?**

# Some Suggestions...

## to hopefully initiate the discussion



- **The basis of all is human communications**
- So what communications ?
  - Internal seminars
    - On whatever topics
    - Virtual for practical reasons
    - But also “agora-like” whenever possible
      - With presentations
      - But why not even with as low prepared as “what do you want to know ?”
        - And we discuss, bombard of questions, interact,...
  - Of course face to face discussion
  - But also some “collaboration cultural background” definition
    - What would we consider that everyone in the Collaboration should know about Geant4 ?
  - Technical notes !
    - We do not have technical notes in Geant4 !
  - And surely other things...
- **So it looks that the “generation transfer knowledge” issue touches upon aspects going well beyond the “senior” to “junior” generations communication.**

# Internal Seminars



- **An idea which was already proposed**
  - With the motivations exposed earlier here
  - **Share knowledge, foster emergence of new ideas, improve collaboration spirit, help with transfer knowledge from “senior” to “junior” generation**
- **Are people convinced of usefulness of such seminars ?**
- **What contents could we consider ?**
  - From “senior” to “junior” generations
    - As presented just before...
  - From “junior” to “senior” generations
    - “junior” generation in good position to “teach” “senior generation” about new technologies –GPU, ML, ...- and new practices –Open Development Model, etc.-
      - Does not mean we adopt all these techniques and practices
      - But feeds the discussion and thoughts, and may lead to evolution
  - But also from “members” to “members”
    - To help with “cultural collaboration background”
    - But also to help to avoid duplicating/triplicating/... technical functionalities because people are unaware they exist already
  - And also “externals” to collaboration
    - To be informed of new trends
    - Eg “differential computing”
- **(quite easy question) Do we share the above considerations ?**
- **(more uneasy question) What do we demand to internal seminars, and how do we shape them ?**

**Opening the discussion !**