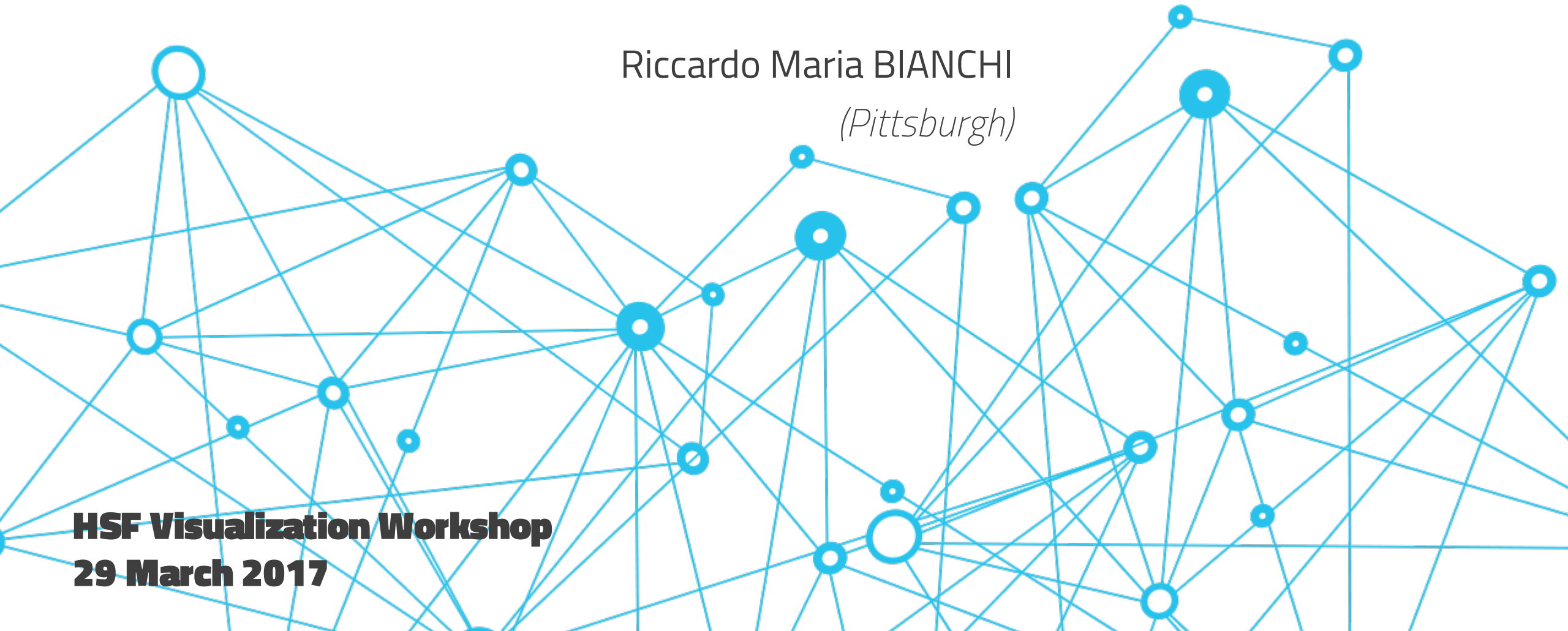


Common approaches: fostering collaboration in HEP

Riccardo Maria BIANCHI

(Pittsburgh)

HSF Visualization Workshop
29 March 2017



The Visualization community

- *Peter Messmer*, who leads the Scientific Visualization department at NVIDIA, said in yesterday's session:
 - *"Scientific Visualization community is small"*
 - *"there are no big interests in building and maintaining software solutions only for this small community"*
- And the HEP Visualization community is even smaller
- At yesterday's session I counted ~ 18 people in the room and ~15 connected remotely.
- Huge success for us!! I think that a large part of our community was there...

Funding HEP SW in the next years

- As I reminded in yesterday's introduction, and as pointed out in Ben Waugh's presentation as well:
 - we have to cope with flat budgets for the next years
 - funding agencies requested to develop common programs when possible, in order to share the funds for the benefit of multiple communities

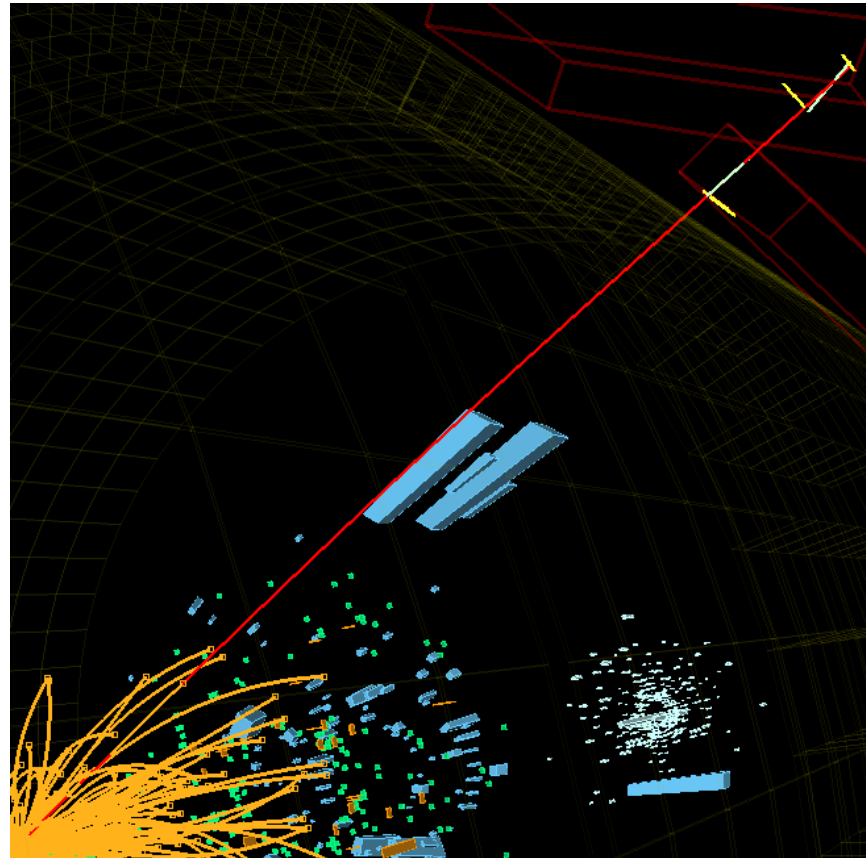
Visualization in HEP

- Visualization is always considered at the very end of the HEP data chain
- But that's not always true: tools which are integrated into the experiments' frameworks are used for detector development or to check and debug data taking, simulation, reconstruction
- Visualization is important in HEP

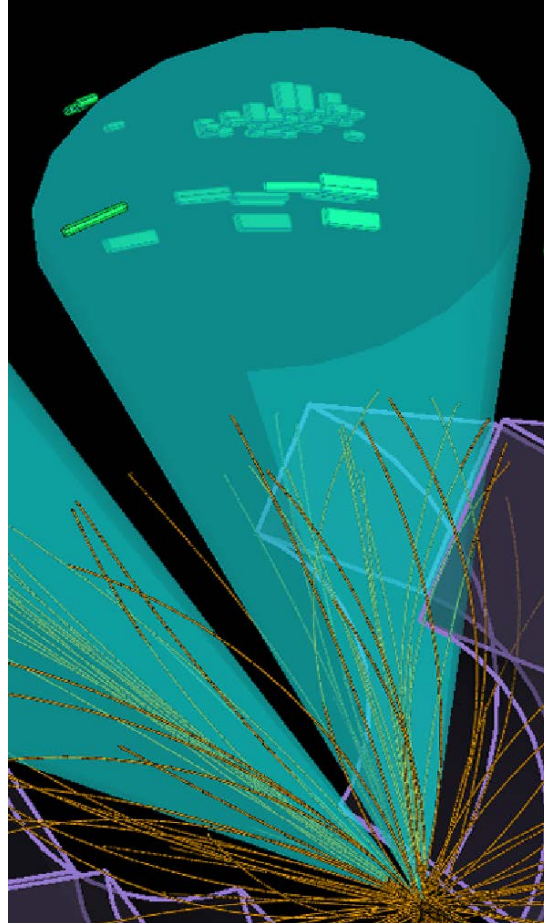
Common needs

- Despite the differences among the experiments (different sub-detectors, data formats, databases, algorithms) we have **many common needs**, specially in Visualization

What is this?

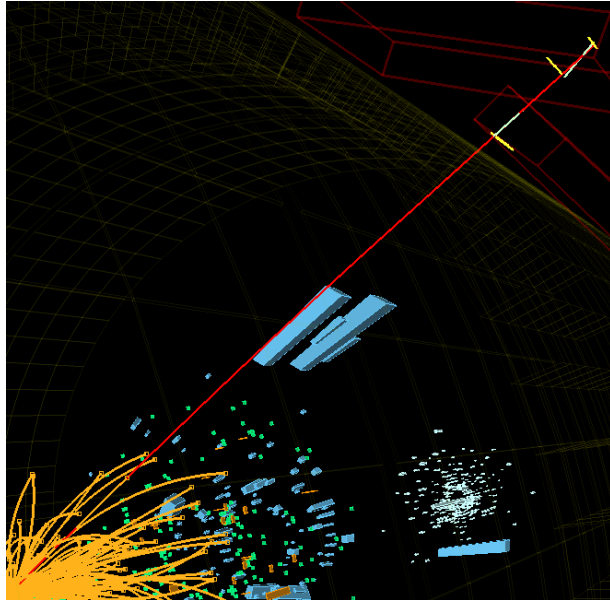


And this?

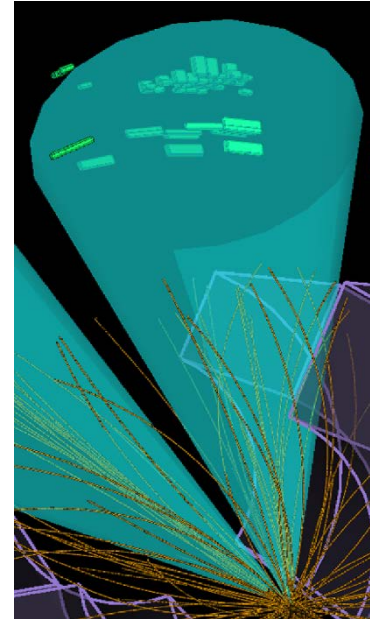


Common needs, common objects

- Despite we are from different experiments, we all answered this



A Track



A Jet

Common base objects

- in whatever HEP experiment :
 - a Track is displayed with a more-or-less curved line,
 - a Jet is displayed with a cone
 - an energy deposit is displayed with a series of boxes
 - and so forth...

HEP software

- In our experiments, for historical reasons, we all used different tools
- Many of our current tools come from the “good old days”, where:
 - computer graphics was at the **beginning**

Coin3D was developed in 2000, based on Open Inventor which was started in 1989...

- **collaborative development** was **rare**

Collaborative development

- Open source tools and community-driven development are the standards today
- Many big **software companies** (IBM, NVIDIA, Google, Oracle, Mozilla, ...) **contribute** with people and code to open-source projects
- Current industry trend is:
 - *If I need a common component, instead of developing one in-house, which then I have to maintain...*
 - *...I contribute to an open-source project and I use it!*

Community-driven software

- Having usually large community behind, bugs are quickly discovered and often quickly resolved
- ...often much faster than in commercial software!

Ever tried to fill a ticket for a Mac OS bug?

- ...and the same for feature request!

Since when the users are expecting that new feature in from Apple or Microsoft?

Outside graphics libraries landscape

qt / qt3d

Watch 13 Star 54 Fork 37

Code Pull requests 0 Projects 0 Pulse Graphs

No description, website, or topics provided.

5,650 commits 11 branches 24 releases 72 contributors

Branch: dev New pull request

Create new file Upload files Find file Clone or download

liangqi Merge remote-tracking branch 'origin/5.9' into dev Latest commit 0ba3824 28 days ago

- config.tests Added FBX geometry loader 2 months ago
- dist Add changes file for 5.8.0 4 months ago

Steve Streeting / ogre

Overview

Last updated 15 hours ago
Website <http://www.ogre3d.org/>
Language C++
Access level Read

9 Branches 21 Tags
592 Forks 762 Watchers

OGRE

Kitware / VTK

Watch 104 Star 617 Fork 398

Code Pull requests 8 Projects 0 Pulse Graphs

Visualization Toolkit <http://www.vtk.org>

63,275 commits 6 branches 60 releases 227 contributors

Branch: master New pull request

Create new file Upload files Find file Clone or download

mathstuf committed with kwrobot Merge topic 'fix-boost-detail-namespace-conflict' Latest commit 0961363 14 minutes ago

- .ExternalData Teach SourceTarball.bash to split source and data tarballs 3 years ago
- Accelerators/Vtkm Merge topic 'add-vtkmCleanGrid-filter' 7 days ago
- CMake Use correct filename for VTKTargets when externally specified. 7 days ago
- Common Merge topic 'fix-boost-detail-namespace-conflict' 14 minutes ago

mrdoob / three.js

Watch 1,957 Star 31,762 Fork 11,307

Code Issues 731 Pull requests 200 Projects 0 Wiki Pulse Graphs

<https://threejs.org/>

webgl canvas svg scene-graph

3 branches 76 releases 786 contributors MIT

New pull request

Create new file Upload files Find file Clone or download

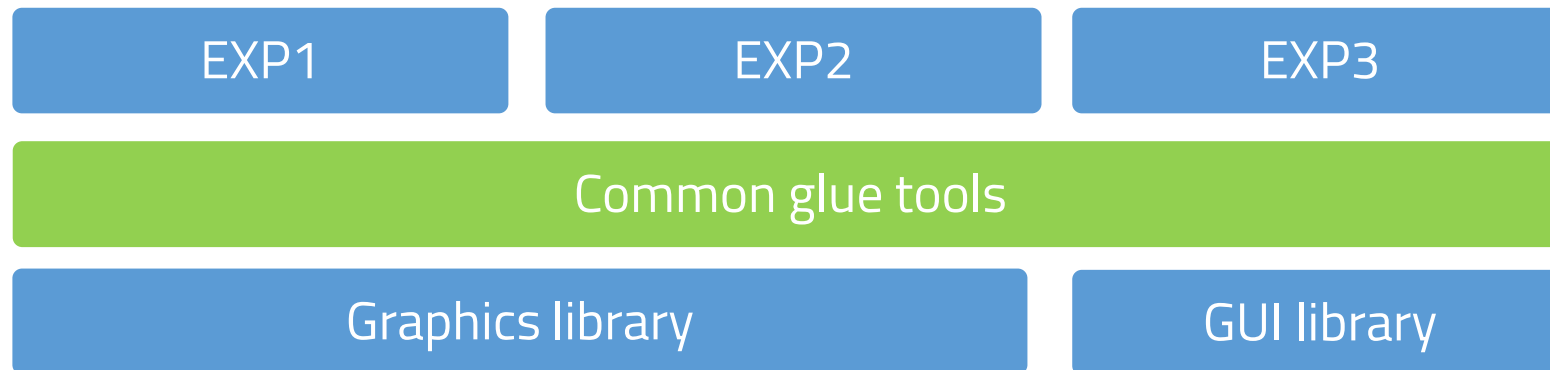
donmccurdy Merge pull request #11081 from donmccurdy/feat-draco-is-version-suppo... Latest commit 3ae54a2 12 hours ago

- Improved ISSUE_TEMPLATE.md 12 days ago
- Updated builds. 2 days ago
- Merge pull request #11061 from BrianMacIntosh/patch-1 17 hours ago
- Editor: Clear codemirror history after setValue. 2 days ago
- Update DRACOLoader and draco_decoder. 13 hours ago
- Geometries: Removed unused variables 2 days ago

Common packages for HEP

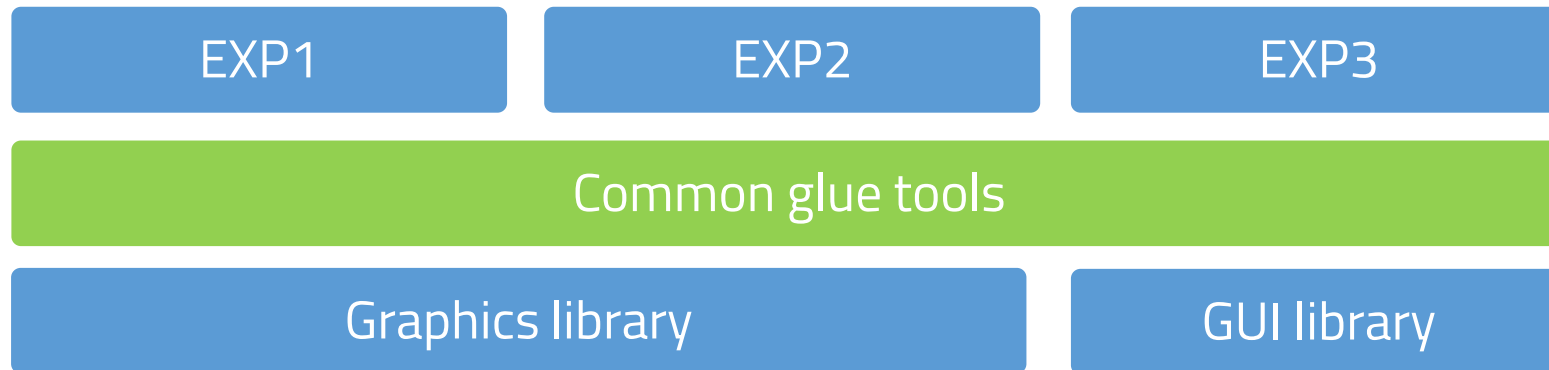
Instead of maintaining our own graphics libraries, our own GUI libraries, our own renderers...

Why not developing **common glue packages** and using **industry standards behind**?



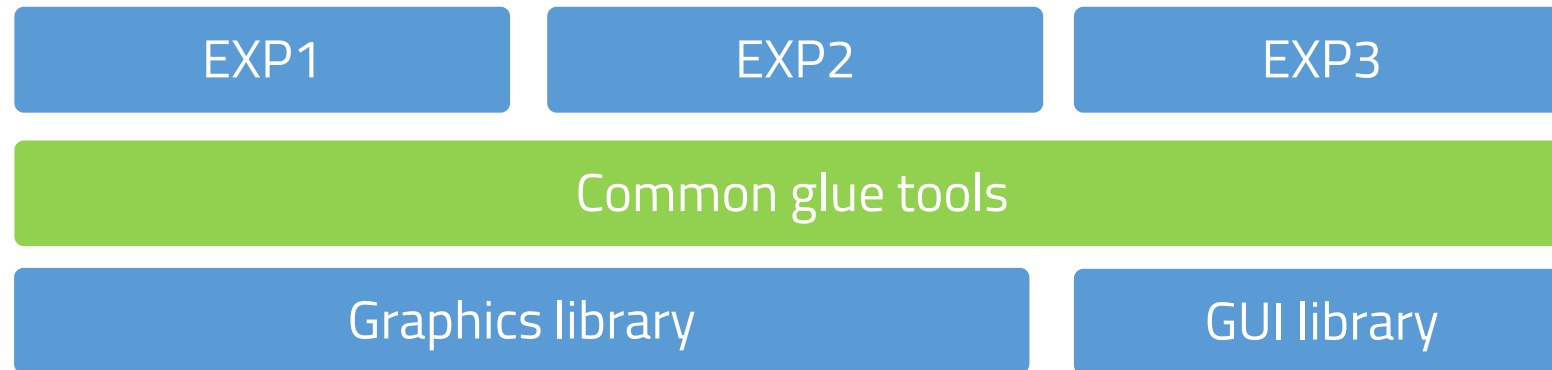
Standard base packages

- We could take advantage of widely developed standard packages and just use them in our software tools



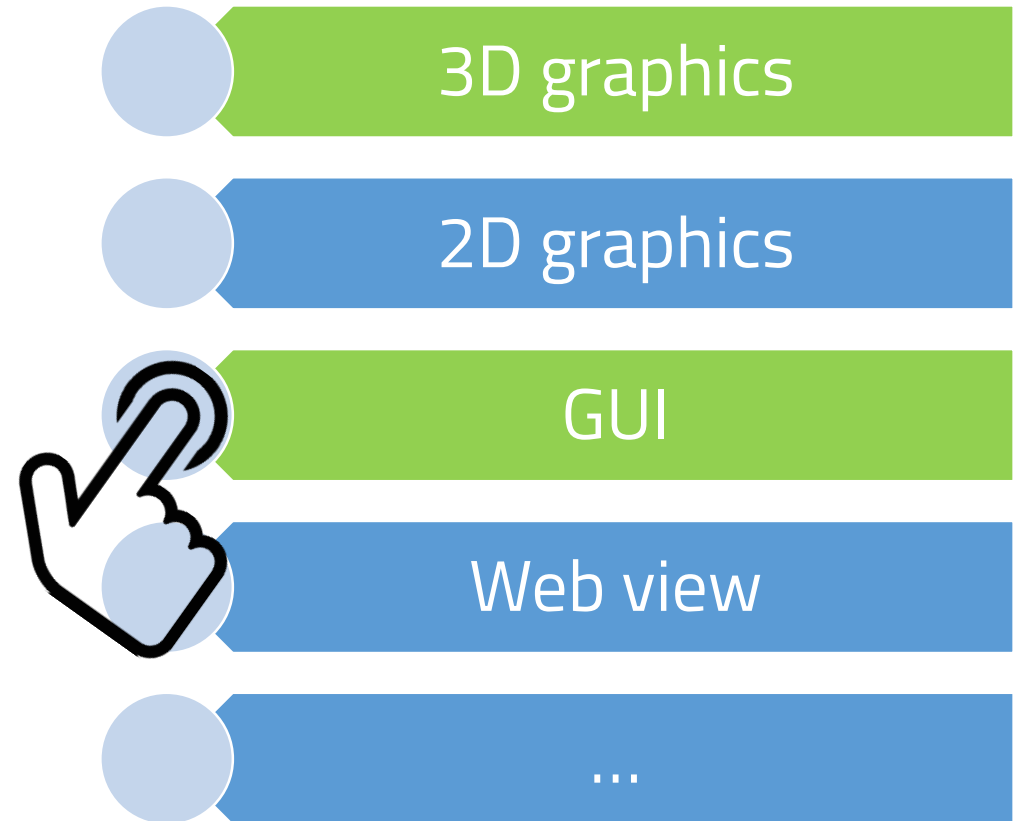
Experiment's targeted layers

- No need for common final software: all experiments should be able to make their own tools to display their own specific data



Pick&Choose

- But I think that experiments should **easily pick&choose** what they need
- ...**without maintaining** the base packages behind



Let's build the collaborative HEP future!

Thanks!

