WHAT IS pROOT?

Doug Benjamin

Duke University

Acknowledgements

- Simone Campana , Nils Krunmack and Alex Madsen for the content of these slides (and their actual code)

Do we need pROOT…

- … like we have pathena and prun?
  - To provide a tool for ROOT based distributed analysis

- Would be able to encapsulate many capabilities we do not have with prun
  - Enable I/O performance settings (e.g TTreeCache)
  - Provide better error handling
  - Better handle distributed environment (timeouts/retries/…)
  - Increased monitoring of the data that users are actually reading

- But many of those would have to be implemented at the event loop level
  - So we would have to provide also an event loop

xAOD reading Classes

- xAOD reading classes are being designed to add functionality needed in our highly distributed environment
  - enabling TTreeCache to improve network data access (wide area or local area)
  - xAOD library will used for ROOT analysis
  - It was agreed to instrument the xAOD classes to report data accesses for popularity
  - The xAOD classes should also report what data is really read by the job. How does this monitoring information get sent to a centralize collection point for further analysis
    - *This information is important to identify which parts of the derived xAOD's are rarely read. (ie write once , read rarely if at all)*

The event loop
(from Simone's slides)

- Some functionalities need to be in the event loop (retries, I/O errors)
  - Today macros are provided to:
    - Create the event loop encapsulating the use ROOT code
    - Submitting it to the Grid via prun
  - We can act here to instrument the event loop
  - PanDA can than handle the information provided
    - Server side retries, exposing monitoring information
  - People will use it if they find an advantage (and if we do it properly, they will)

- The same functionalities should be implemented in Athena

Solution to the event loop issue

- The **EventLoop** package (written and maintained by Nils Krumnack) – provides an event loop for processing xAOD’s.
  - Code has existed for some time
  - Being taught in the software Tutorials
  - Is becoming the de-facto ATLAS wide analysis standard
- EventLoop Grid driver extension (written and maintained by Alex Madsen) provides linkage between EventLoop and PanDA
  - In production for some time
  - Works with JEDI already
  - Has detailed error reporting including which errors can be retried or not. This information needs to propagate into JEDI

Open questions (ie discussion topics)

- Exactly what information should collected by the xAOD reader classes?
- How do we get the xAOD reader class authors to include the monitoring data?
- How is this monitoring data collected? Does EventLoop send the information to Activie MQ collector somewhere?
- How much monitoring information is too much?
- How and who will analyze the monitoring information?
- Does EventLoop have enough error handling currently? What about retries within EventLoop in case of errors? Does retries within EventLoop really make sense?
- Does the US take an expanded role in all of this?

The End

- Time for discussion.