SoLID Software Update & Responses to Recommendations

Ole Hansen

Jefferson Lab

SoLID Collaboration Meeting
August 26, 2016

Past 12 Months: Deciding on a Software Framework

Framework	Pros	Cons
art (FNAL)	• Large user base • Developed by experts • Very good documentation • Modern • ROOT6 support • Best match to our requirements	• Not multi-threaded, not distributed (but multi-threading planned) • Heavy binary installation by default • In-house build system • Somewhat complex
FairROOT (GSI)	• Familiar ROOT environment • Large user base (incl. EIC a.t.m.) • Distributed processing extension (experimental) • Good built-in simulation support	• Absent documentation • Poor API definition • Old code base • Existing code tends to be a mess • Single-threaded (unlikely to change) • Heavy dependency requirements
Fun4All (PHENIX)	• Lightweight • Well-tested, proven performance • Familiar ROOT environment	• One-man project • Very PHENIX-centric • Absent documentation • Very old code base • Many missing standard features • Single-threaded (unlikely to change)
JANA (JLab Hall D)	• Multi-threaded • Lightweight • Local expertise	• Small user base • Too many technical limitations • In-house DST format (HDDM)
Clara (JLab Hall B)	• Multi-threaded and distributed • Local expertise	• Small user base • Java based • Very complex • Performance concerns • In-house DST format (EVIO)

Past 12 Months: Deciding on a Software Framework

Framework	Pros	Cons
art (FNAL)	• Large user base • Developed by experts • Very good documentation • Modern • ROOT6 support • Best match to our requirements	• Not multi-threaded, not distributed (but multi-threading planned) • Heavy binary installation by default • In-house build system • Somewhat complex
FairROOT (GSI)	• Familiar ROOT environment • Large user base (incl. EIC a.t.m.) • Distributed processing extension (experimental) • Good built-in simulation support	• Absent documentation • Poor API definition • Old code base • Existing code tends to be a mess • Single-threaded (unlikely to change) • Heavy dependency requirements
Fun4All (PHENIX)	• Lightweight • Well-tested, proven performance • Familiar ROOT environment	• One-man project • Very PHENIX-centric • Absent documentation • Very old code base • Many missing standard features • Single-threaded (unlikely to change)
JANA (JLab Hall D)	• Multi-threaded • Lightweight • Local expertise	• Small user base • Too many technical limitations • In-house DST format (HDDM)
Clara (JLab Hall B)	• Multi-threaded and distributed • Local expertise	• Small user base • Java based • Very complex • Performance concerns • In-house DST format (EVIO)

Framework Choice

- Considering many factors, *art* appears to be the overall most suitable software framework for SoLID that is readily available at the moment
- Current version (2.0.3) installed on JLab/ifarm
- Testing and prototyping underway
- First priority should be to port the existing simulation chain into *art*.

Draft Task List for *art*@SoLID

Task	Status	Time est (FTE)
Geometry Service	Started (25%)	1 month
Conditions Database Service (with CCDB backend)	Not started	2 weeks
Producer module prototype interfacing with Geant4, similar to artG4	Not started, but working example exists	2 weeks
Draft data model (digits, hits, clusters, …) and corresponding data classes. Will be refined during algorithm development	Not started	1 week
Port/implement algorithms • Generators • Digitization • Clustering • Track pattern recognition • Track fitting • PID (Calos, Cherenkovs, ToF, MRPC) • … more …	Not started, but some algorithms exist and can be ported	(extensive)
Decide on preferred software packaging, platform support (e.g. Mac OSX?), build system, distribution etc.	Started (10%)	1 month (existing system is complex)
EVIO decoder, mapping, corresponding databases	Not started, but have working examples in Hall A/C analyzers & GlueX code	1 month

Resources, Documentation

- The *art* web site:

  `http://art.fnal.gov`

- *art* workbook (new version July 2016):

  `http://art.fnal.gov/wp-content/uploads/2016/03/art-workbook-v0_91.pdf`

- August 2015 software workshop (many informative talks)

  `https://indico.fnal.gov/conferenceDisplay.py?confId=9928`

- Tutorial on the *art* test installation at JLab

  `http://hallaweb.jlab.org/12GeV/SoLID/meeting_coll/2016_05/Hansen-SoLID-CollabMeeting-Software-2016-05-06.pdf`

- Wiki (lots of developer information)

  `https://cdcvs.fnal.gov/redmine/projects/art/wiki`

Director's Review Recommendations I: End-to-End Framework

- **Recommendation:** "End-to-end simulations with realistic subsystem responses and material budgets, and complete track finding and reconstruction should be developed."
- **Recommendation:** "The development of a simulation framework with realistic reconstruction and analysis should be pursued with high priority and increased resources."
- **Recommendation:** "The collaboration is strongly encouraged to develop an end-to-end realistic simulation and reconstruction to further optimize cost and physics reach and derive clear performance requirements for the individual subdetectors."
- **Recommendation:** "Having functional simulation and reconstruction routines as soon as possible should be a high priority in the software effort. Such software will pay off many times over in experimental design and avoiding pitfalls."

Responses I: End-to-End Framework

- We have (tentatively) chosen the *art* famework from Fermilab as a basis for long-term SoLID simulation, reconstruction and analysis software development
- Testing and prototyping to gain expertise with *art* is currently in progress
- A detailed software design document is being drafted and expected to be available by the end of 2016
- Porting of existing simulation, digitization and reconstruction algorithms to this framework will commence later in 2016
- Long-term development from approximately mid-2017 onwards will be done within *art* whenever possible

Director's Review Recommendations II: Manpower

- **Finding:** "Consultation with appropriate people from the other halls would be useful to get a more accurate estimate of software needs, including manpower."
- **Recommendation:** "Compare the resource levels you have assumed in some key areas (particularly in software, . . .) to make sure the estimates align with other similar projects or there is a good reason they do not."

Responses II: Manpower

- We are (keenly) aware of the high manpower needs for software development. The estimates made by the other halls (30-40 FTE-years) seem roughly accurate for SoLID as well.
- (Status of "consultations with other halls"?)
- 3.5 (?) FTE postdoc positions have been requested in our recent pre-R&D funding proposal

Director's Review Recommendations III: Tools & Collaboration

- **Finding:** "Early exploration of the tools available at Jefferson Lab that can handle the data at the expected scale of SoLID will be crucial in minimizing false starts in software development."
- **Recommendation:** "Closer communication with the other JLab experiments and the JLab computing center is strongly encouraged."

Responses III: Tools & Collaboration

- We have been in active communication with the JLab computer center regarding future computing needs for SoLID. Based on current trends, handling of data volumes at the expected scale of SoLID, *viz.* 5-10 PB/year, is already fully managable at JLab today and will likely be routine at the time SoLID runs.
- We are investigating the suitability of the existing JLab workflow management tools (SWIF) for SoLID computing. We are also aware of, and will probably investigate in the future, alternative workflows that have proven successful in HEP, in particular "analysis trains" employed at LHC, RHIC and elsewhere. In the long run, it would likely be beneficial if SoLID software supported distributed and/or grid computing. We will keep this possibility in mind. Any advanced data processing capabilities would be developed in close collaboration with the computer center and the other halls, which are in part already exploring distributed computing and would certainly all benefit from a common approach.
- Substantial data for GlueX have just begun to arrive in 2016. CLAS12 is expected to go into production mode in 2018. Further, the Hall A SBS program, which will also produce multi-PB data sets, will commence in 2019. The experiences of these groups, as they emerge, will inform future decisions we may have to make for SoLID software development.