

LCG Applications Area Internal Review Response (preliminary and brief version) (main points are on last slide)

Torre Wenaus, BNL/CERN
LCG Applications Area Manager

<http://lcgapp.cern.ch>

PEB

Dec 2, 2003



Applications Area Internal Review Oct 20-22

- ◆ Technical review of applications area software and its integration into the experiments, particularly POOL and SEAL
 - ◆ Design, implementation, experiment experience, performance, coherence, problems/risks, evolution
- ◆ Committee members from experiments, ROOT, EP, people involved in AA software integration
- ◆ Final report Nov 12, now being digested and actions formulated
 - ◆ We are assembling a written response to the report
- ◆ Very constructive and useful recommendations
- ◆ Partial and preliminary responses, actions on following slides
 - ◆ Blue = recommendations already followed or already policy
 - ◆ Red = action underway or planned for the near term
 - ◆ *Italics* = we agree and will have to work out how to do this
 - ◆ **Bold** = main issues identified by committee



Internal Review – SPI main points

- ◆ SPI services not used consistently – **central librarian needed**
 - ◆ EP/SFT acting to put in place a highly qualified central librarian
- ◆ SPI should use IT services – test and adopt IT-CVS ASAP
- ◆ *Interaction of SPI with other LCG areas should be clarified; work with GDA on distribution (pacman)*
- ◆ Savannah scalability, maintain GNU collaboration
- ◆ Automated nightly testing. **Back up policies with tools**
- ◆ **Nightly build model insufficient. Provide tools for immediate build on the supported platforms. Long term NICOS maintenance.**
- ◆ **Build system should be simpler and more consistent**
- ◆ **Proceed with autotools based build investigation. Involve experiments in the investigation and coordinating any migration. Decide on and use one tool. Ensure optimal support for experiments.**
- ◆ *Provide SCRAM and CMT configuration files for experiment use.*



Internal Review – POOL

- ◆ User documentation. Particularly public API, ref<T>, dictionary, ROOT I/O features (not) supported by storage manager
 - ◆ First version of comprehensive UG released with POOL 1.4 – for constant iteration and improvement in subsequent releases
- ◆ *Address support for ROOT schema evolution on few-month timescale*
- ◆ *POOL files must be fully browsable with ROOT (including ref<T>)*
- ◆ Close experiment interaction on new needed features
- ◆ Optimize performance on realistic use cases; storage manager ROOT performance should be within few percent of ROOT itself
- ◆ More realistic, larger scale test cases, using experiment data models
- ◆ Catalogs should provide authentication/authorization. Address in AF and ARDA
 - ◆ We see this as coming from middleware, not a POOL concern
- ◆ **All collections should be browsable, including within ROOT. Must be usable concurrently in a parallel environment.**
 - ◆ Collections expected to be an ARDA AA focus.



Internal Review – SEAL

- ◆ The measure of SEAL success is SEAL usage, in other projects and the experiments. SEAL must do more to engage the experiment community to get its offerings integrated, evaluated and adopted. More focus on customers.
 - ◆ Emphatic agreement.
- ◆ Foundation libraries: Tutorials, user guides, help. *Review external package dependencies. Work closely with experiments and ROOT to try to converge on one plug-in manager.*
- ◆ Mathlibs: Concern over uncertain future of Minuit, GSL, CLHEP. Adopting/adapting third party libs (eg GSL, Boost) cost-effective. Careful testing required. Specific HEP functions needed. Provide coherent set of libraries with: dictionaries; interactive use; persistency where required; build/install support on multiple platforms.
 - ◆ Planning of revitalized mathlib effort underway now, with strong EP/SFT support and drawing on GSL, Minuit, ROOT, CLHEP,...



Internal Review – SEAL (2)

- ◆ **Dictionary: Size is an issue. *Impact on performance should be evaluated by experiments. LCG, ROOT dictionaries should merge, by a firm date.***
 - ◆ Good agreement and progress on this issue since the review. Will incorporate in 2004 workplan now being developed.
- ◆ **Framework services: *Must engage the experiments if it is to avoid irrelevance. Discuss with the experiments their interest and set goals, priorities and manpower accordingly.***
 - ◆ Emphatic agreement.
- ◆ **Scripting: Boost vs. SWIG. Discuss with experiments. Avoid development until need arises. Proceed with PyLCGDict as complementary and immediately useful (with PyROOT). *Seek feedback from experiments on python usability in interactive analysis.***
 - ◆ Boost vs SWIG settled (on Boost) in AF. Proceeding with PyLCGDict.



Internal Review – Simulation

- ◆ Generator services: *Set up a testbed for comparison of event generators. Support two event data formats, low-volume (e.g. XML) and high-volume (POOL/ROOT), with data structures supporting MCtruth-data comparison.*
 - ◆ Event data format plans recently developed are consistent with this.
- ◆ Physics validation: *Build a common physics list minimizing parameters to be tuned. Experiments should contribute more; it is an excellent opportunity to work on a physics project.*
- ◆ **Generic framework: Develop the VMC as abstract interface to simulation engines and the main tool in medium/long term. Use FLUGG to support Fluka via G4 geometry in the short term. Set program, priorities and manpower through discussion with experiments.**
 - ◆ Recently decided in AF: use FLUGG now, move to VMC next year when ready to try out and evaluate VMC and geometrical modeller, with the goal of subsequent collaborative development
 - ◆ Initial target is physics validation; use that experience in developing a long term plan for full detector simulation



Internal Review - PI

- ◆ For interactive analysis, direct use of ROOT (via PyROOT or CINT) is the recommended solution.
 - ◆ Consistent with AF decision.
- ◆ ROOT must be able to access data stored with POOL.
- ◆ **PI project should be rediscussed in light of ARDA.**
 - ◆ Planning ARDA activity in the applications area is just beginning. (We will have time for it when this review is over!)



Internal Review – ROOT

- ◆ From the report's conclusion...
 - ◆ The analysis of the specific comments on each sub-project shows that most of the projects interoperate with ROOT. In order to integrate ROOT in the architecture and to optimize the performances of the services the committee recommends that the technical collaboration between the different sub-projects and ROOT evolves from a client/provider mode to a modular cooperation as agreed in the blueprint RTAG.
- ◆ How I would put it...



ROOT role and relationship

- ◆ Blueprint established a user/provider relationship
- ◆ Basically worked well – productive cooperation – but the internal review triggered recognition that the relationship needs strengthening given ROOT's role:
- ◆ ROOT is a central element of the LCG software: it is today and will be for the foreseeable future
 - ◆ The basis of the event store for all experiments
 - ◆ The principal analysis tool used by all experiments
 - ◆ Used directly by ALICE as the basis of their framework
- ◆ Rene has made a well-received proposal to the AF on a new relationship aimed at convergence and coherence
 - ◆ With specific objectives, beginning with a common dictionary and extending to math libraries, POOL-ROOT interoperation, simulation (VMC, geom modeller), SPI usage and others
- ◆ Applications area meeting presentation on Dec 10 presenting Rene's proposal and discussing actions undertaken and planned



My View of the Main Points (Random Order)

- ◆ New ROOT relationship, as expressed initially in development of
 - ◆ Concrete plan to converge on common dictionary
 - ◆ Common proposal on math libraries
- ◆ Improve integration support; Developer-experiment associations
- ◆ LCG/AA central librarian in SPI
 - ◆ EP/SFT pursuing this as a priority
- ◆ gmake+autoconf build system alternative: evaluation and decision
 - ◆ Support SCRAM, CMT configurations whatever the in-house choice
- ◆ POOL schema evolution
- ◆ SEAL engagement of user community
- ◆ Generic simu framework program leveraging FLUGG now and evaluating VMC when ready
- ◆ ARDA program and its impact/relation to existing (PI, POOL collections, ...)

