SEAL Project Status and Plans

Application Area Meeting
March 3rd 2003
P. Mato / CERN

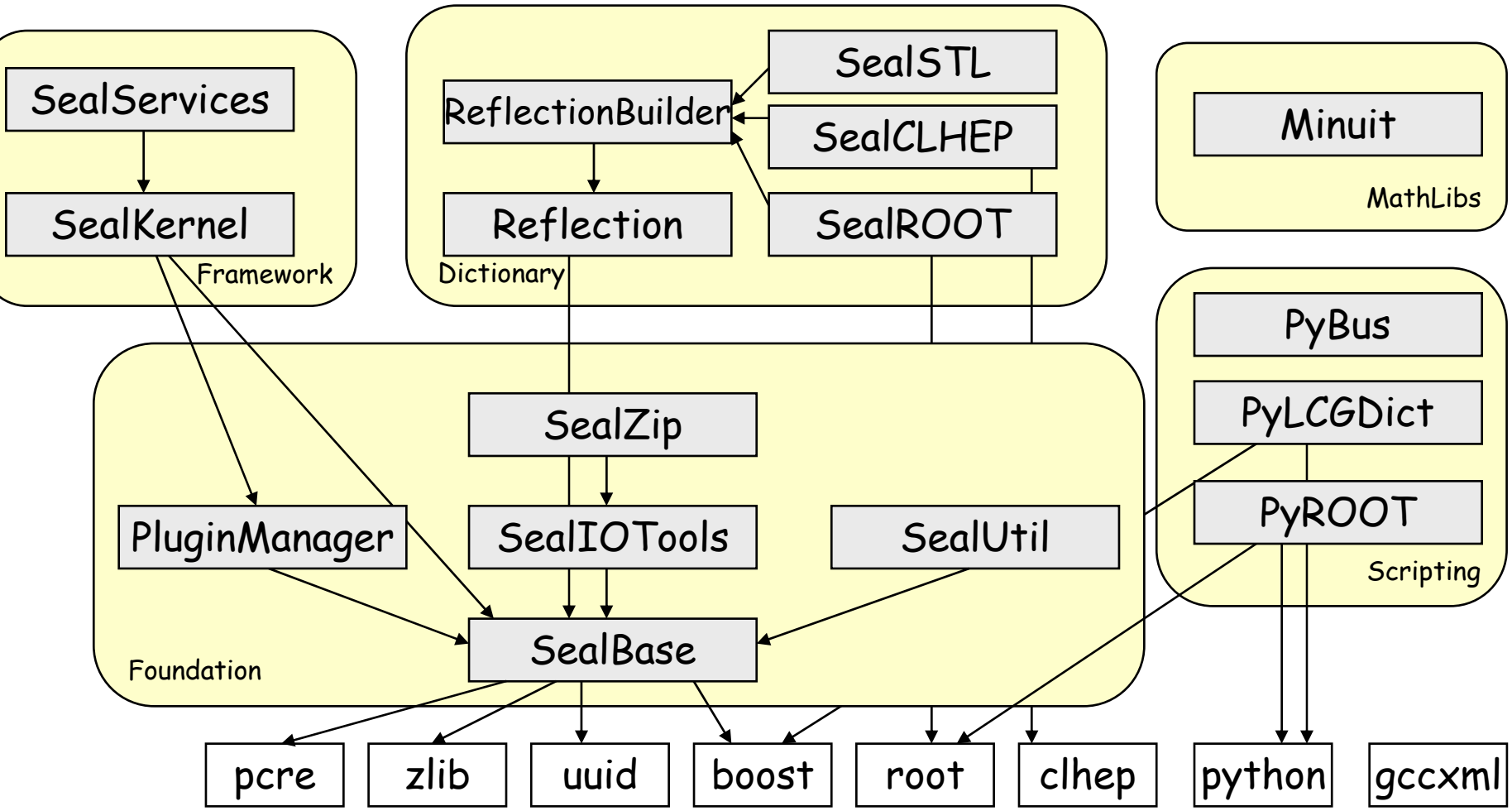
Contents

- Current Software Status
- Project Plan for 2004
- Resources
- Conclusions

SEAL Work Packages

Foundation	Foundation and Utility Libraries and Plug-in Manager
MathLibs	Math Libraries Support and Coordination
Dictionary	LCG Object Dictionary
Framework	Component Model and Basic Framework services
Scripting	Scripting Services
Documentation	Education and Documentation

Packages and Dependencies (v1.3.3)



Summary Status

- SEAL 1.3.3 is out since mid-January
  - Should be sufficient for the scheduled experiment data challenges
- Supported Platforms
  - RH73 gcc3.2, Windows vc++7.1
  - icc 8.0 is basically ready waiting for external packages for icc 8.0
- SEAL has delivered a number components that constitutes the basic foundation and utility libraries and object dictionary
  - The main "client" has been POOL
  - Currently being integrated into experiments' frameworks
- The first version of the Component Model and Framework services available
  - Must engage experiments to seek feedback before developing more services
- Scripting based on Python
  - Boost.Python and PyLCGDict recommended to provide Python bindings

SEAL Work Plan

- Short term plans
  - Goals for release 1.4.0 in the time scale of few weeks
- Longer term plans
  - Goals the next 12 months
  - Inputs taken into account
    - » Needs of the LHC experiments in the domain of core software foundation and basic services
    - » The Blueprint RTAG recommendations
    - » Recommendations of the LCG-AA Internal Review
    - » Wish from the line management to define a more coherent collaboration model with the ROOT project

LCG/ROOT Collaboration Model

- Evolve the current *user-provider* relationship to a more coherent one
- The goal is to establish a more peer-to-peer relationship
  - In some cases LCG uses ROOT (e.g. POOL, PyROOT, etc.)
  - In other cases ROOT uses LCG (e.g. New Dictionary, etc.)
- Two areas have been identified as initial attempts for common LCG/ROOT developments
  - Dictionary
  - Mathematical Libraries

Plans for release 1.4.0

- Foundation Work Items
  - Remove unnecessary dependencies in external packages. Study the feasibility of eliminating dependencies to external packages like pcre, uuid, re
  - Improvements in the test suit. Merging the small individual tests into bigger executables.
  - Simplification of the PluginManger. This is to reduce the number of classes the user has to provide.
- Framework Work Items
  - Help POOL to adopt component model. Incorporate all the changes needed for POOL in the component model. This work is needed to simplify the POOL code related to plugin management by basically halving the number of required classes and facilitating the component configuration.
  - .INI style configuration service. Finalizing the development of a new implementation of the configuration service using this syntax.
- Dictionary Work Items
  - Dictionary service. Finish the development of a service to interrogate about any class that is not found already loaded.

Plans for release 1.4.0 (continued)

- Scripting Work Items
  - Introduce the Python courses and tutorials as part of the SEAL release.
  - New implementation of PyLCGDict. Finish the ongoing development of the new version and replace the old one.
  - Make PyROOT not dependent of Boost.Python. Study the possibility to be added as part of the ROOT distribution.
- MathLibs Work Items
  - Minuit major cleanup. Take the opportunity of a major release to perform a major clean-up and performance optimization.
  - Development of a test suit for GSL/ROOT/NagC.
- Documentation Work Items
  - Workbook. First implementation of the SEAL workbook.
  - Development of topical User's Guide (e.g. PyLCGDict, SealBase, …)

SEAL 2004 Program of Work

Foundation Plans

- Not foreseen major new developments in this work package
  - The work that is needed is mainly to educate users by developing and setting up tutorials, user-guides and coaching developers
- Development of Tutorials
- Development of a set of web pages describing and categorizing all the available foundation classes

Framework Plans

- We must engage the experiments if we want to avoid irrelevance
  - The objective is to integrate SEAL component model into the existing Gaudi/Athena framework and evaluate its costs and benefits
  - Before new functionality is added we need to get the feedback from the experiments
  - The development of other services (i.e. Whiteboard) is put on hold until there is a firm commitment from at least 2 experiments
- Provide the necessary help from the project to integrate the existing SEAL component model into Gaudi/Athena
- Provide the necessary help for POOL to adopt the component model

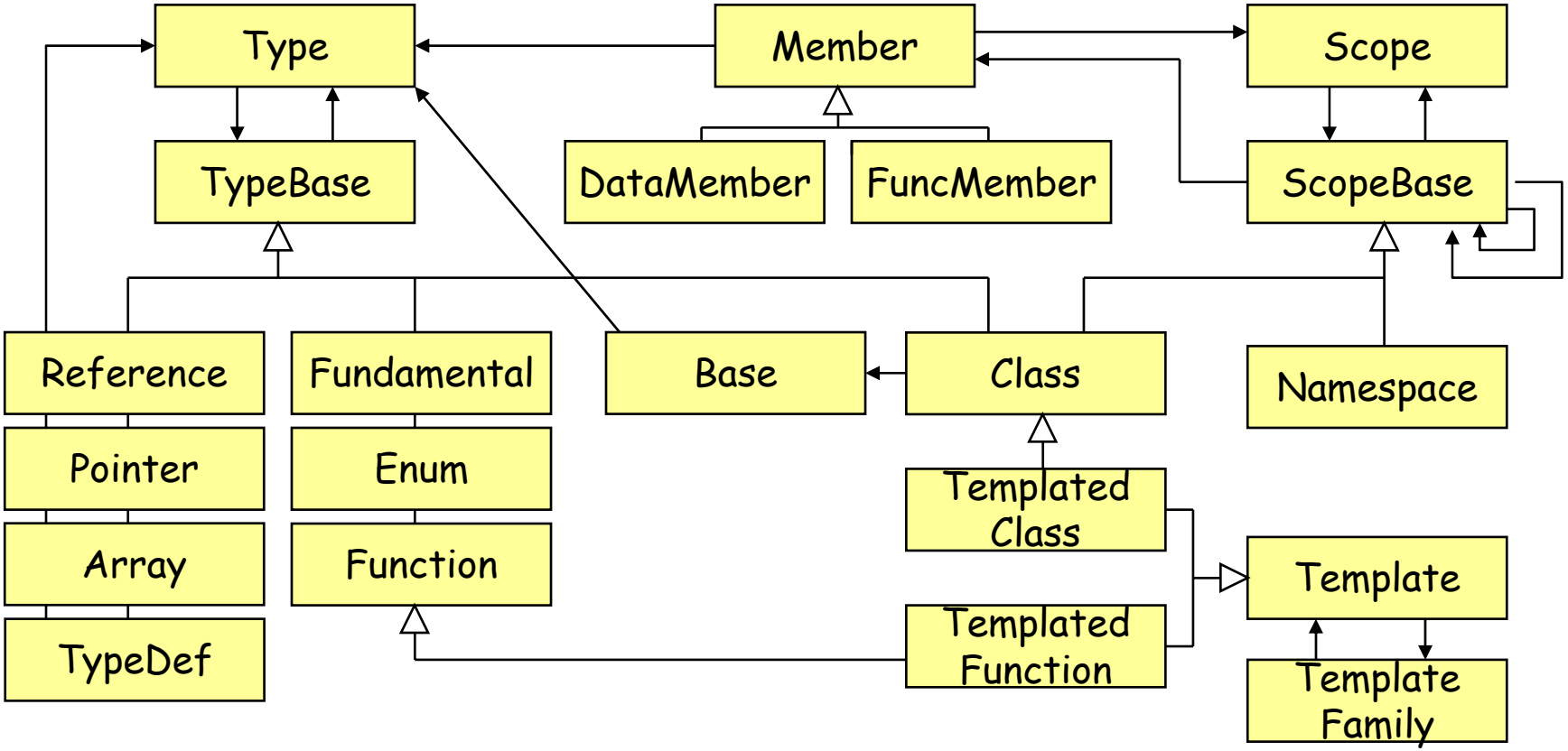
Scripting Plans

- The goal is to seek feedback from the experiment physics community in the usability of python in interactive analysis
  - This work should be coordinated with the other projects (POOL, PI, ARDA,…)
  - The idea would be to identify physicists that would like to act a guinea-pigs.
- Continue the python tutorials and provide help to the user and developer community
- Provide access to key services in such a way that Physicists can asses Python in a useful context, rather than as an abstract language choice in isolation from their real problems

Dictionary Plans

- New Reflection API to overcome current limitations
  - Development done in collaboration with ROOT team. Iterate the current proposal until an agreement is reached.
  - http://cern.ch/seal/snapshot/work-packages/dictionary
- Provide reference implementation of the agreed API as baseline.
- Adapt code generation tools (gcc_xml based) to produce new dictionary descriptions.
- Adapt existing ROOT reflection classes to the new common dictionary.
- Develop an implementation using available CINT data structures with the goal to eliminate the need of having the LCG<->CINT gateway (POOL).

Reflection Model



MathLibs Plans

- The purpose is to provide a coherent Mathematical Library to the end-users
  - Coordinate the activities with ROOT, bringing the needs of LCG and ROOT together, trying to avoid maintenance and support of various mathematical libraries providing similar functionality
  - The goal is to share a common mathematical library between ROOT and the rest of LCG activities and experiments. ROOT should be able to build layered functionality based on this common library.
- A major requirement is to use the same basic Mathematical Library in all environments
  - Directly in C++ within the experiment reconstruction or simulation programs
  - During the analysis phases from an interactive environment, using either Python or ROOT/CINT
- Development done in collaboration with the LHC experiments

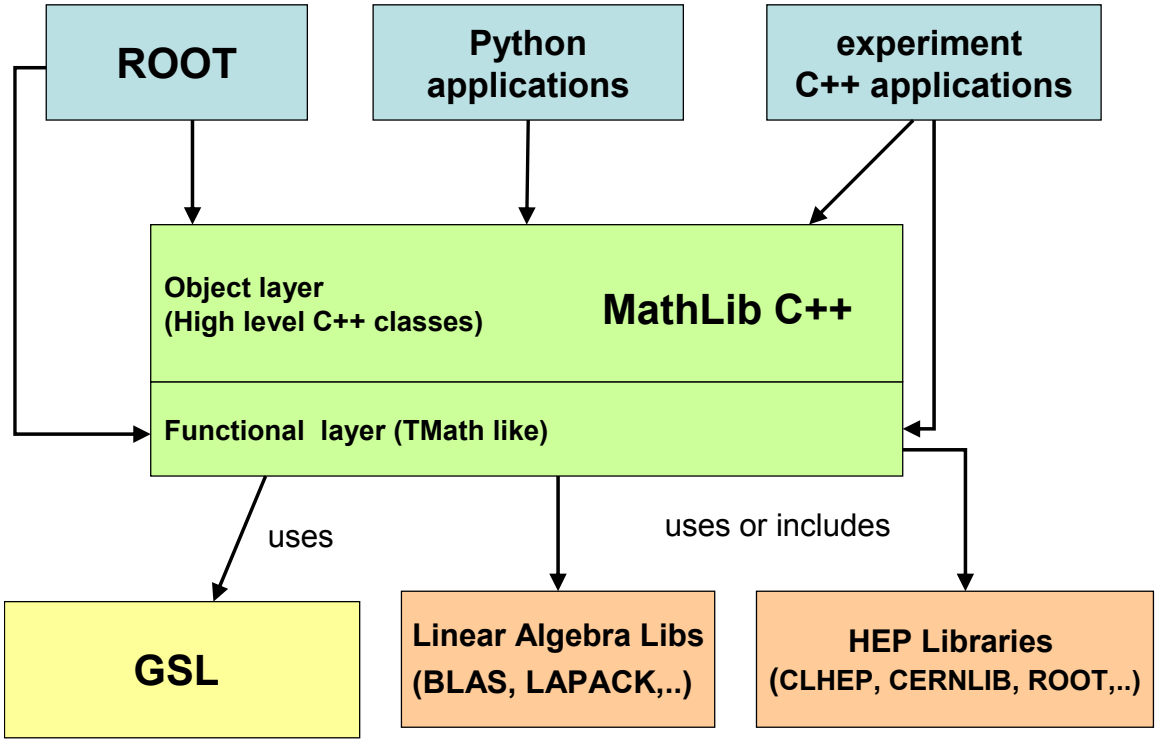
MathLibs: Work Items

- MathLib Web Site
  - Inventory of most common Mathematical functions and algorithms used by the HEP community
  - High quality user documentation should be provided as Web pages describing the palette of functions and algorithms (FAQ, recommendations, references)
  - A cross reference table with links to and from CERNLIB list of routines
- Evaluation of GSL
  - It is expected that the majority of needed mathematical functions is provided by GSL
  - To be re-assured of the quality of the library, we need to study its functionality, numerical stability, performance, accuracy, etc
  - The result of these studies will be a validation and a test suite complementing the one supplied with the GSL distribution

MathLibs: Work Items (continued)

- C++ MathLib
  - Development of a common Mathematical Library
  - Provide a thin wrapper layer in C++ for the set of chosen functions and algorithms (GSL, CERNLIB, ROOT, CLHEP, etc.)
  - Develop a higher layer of C++ classes describing mathematical properties of functions and related algorithms, such as derivatives, integrals, function operation and compositions, etc…
  - Produce dictionaries using the LCG dictionary and CINT for allow interaction from Python and ROOT/CINT
  - Study possible solutions and establish a plan for new developments of a C++ linear algebra library

MathLib C++


ROOT
Python applications
experiment C++ applications
Object layer
(High level C++ classes)
MathLib C++
Functional layer (TMath like)
uses
uses or includes
GSL
Linear Algebra Libs
(BLAS, LAPACK,..)
HEP Libraries
(CLHEP, CERNLIB, ROOT,..)

MathLibs: Work Items (continued)

- CLHEP
  - Continue to provide support for CLHEP (installation, consultancy, etc.)
  - Collect requirements from experiments and discuss them with CLHEP editors
  - Enhance and update existing tests and produce a validation and test suite
- Fitting and Minimization
  - Complete the current development of the Minuit C++ library with full functionality (Simplex, Scan, Contours, etc…)
  - Produce complete Minuit documentation
  - Prototype of fitting and minimization framework using existing code
  - Integrate prototype in existing analysis tools and experiment frameworks

Documentation Plans

- Produce a document describing the process for evaluating, selection criteria and integrating external software
  - Circulate proposal and obtain agreement
- We plan after SEAL 1.4.0 has been released to start a serious campaign in documentation for all aspects of SEAL
- This will be by providing user guides to complement the existing reference documentation and by enhancing the existing workbook

Required resources

Work package	Work Item	FTEweek	FTE
Foundation			**0.4**
	Modifications for 1.4.0	4	0.1
	External Software Guidelines	4	0.1
	Development of Tutorials		0.1
	Development Web Pages		0.1
MathLibs			**3.1**
	Modifications for 1.4.0	4	0.1
	Mathlib Web Site		0.5
	Evaluation of GSL		0.5
	C++ MathLib including Linear Alg.		1.0
	CLHEP support		0.3
	Fitting and Minimization		0.7
Framework			**0.3**
	Modifications/Additions for 1.4.0	4	0.1
	Integration into Gaudi/Athena	4	0.1
	Help POOL adopt component mode	4	0.1
Dictionary			**1.0**
	Additions for 1.4.0	2	0.1
	New Reflection API	8	0.2
	Reference implementation	8	0.2
	Adapt generation tools (gccxml)	4	0.1
	Implementation using CINT structur	10	0.3
	Adapt ROOT reflection classes	8	0.2
Scripting			**0.7**
	Modifications/Additions for 1.4.0	6	0.2
	Python tutorials		0.2
	Help to Physicists		0.3
Documentation			**0.3**
	Documentation for 1.4.0	4	0.1
	New documentation		0.2
Total			5.8

Available people (~6 FTE)

Foundation	**Lassi Tuura**, Lorenzo Moneta, Massimo Marino, Radovan Chytracek
MathLibs	**Lorenzo Moneta**, Matthias Winkler, Marte Hatlo
Dictionary	**Stefan Roiser**, Pere Mato
Framework	**Radovan Chytracek**, Lassi Tuura, Pere Mato, Massimo Marino
Scripting	**Jacek Generowicz**, Pere Mato, Wim Lavrijsen, Massimo Marino
Grid	
Documentation	Jacek Generowicz

New resources could be made available (from CMS, LHCb) for the development of the new MathLib C++

Proposed Milestones

- External software guideline document (31st May 2004)
- Workbook for SEAL including complete HowTos and examples for each SEAL component (30th June 2004)
- New Dictionary API and reference implementation (30th June 2004)
- MathLib project Web (30th June 2004)
- First version of the C++ MathLib package (30th September 2004)

Conclusions

- Draft Planning Document available at
  http://seal.web.cern.ch/seal/documents/SEAL_Program_of_Work_20040302.doc
- SEAL 1.4.0 should be ready in few weeks
  - No major new functionality. Natural evolution.
- The proposed work plan implements most of the internal review recommendations
  - Help experiments and end-users to integrate SEAL functionality
  - Develop a new collaboration model with ROOT project
  - Get feedback from the experiment physics community in usability of python in interactive analysis
- Eager to stop planning and start working