SEAL Project Status

SC2 Meeting
16th April 2003
P. Mato / CERN

Contents

- Project Overview
- Current Status
- Work Plan
- Resources
- Summary

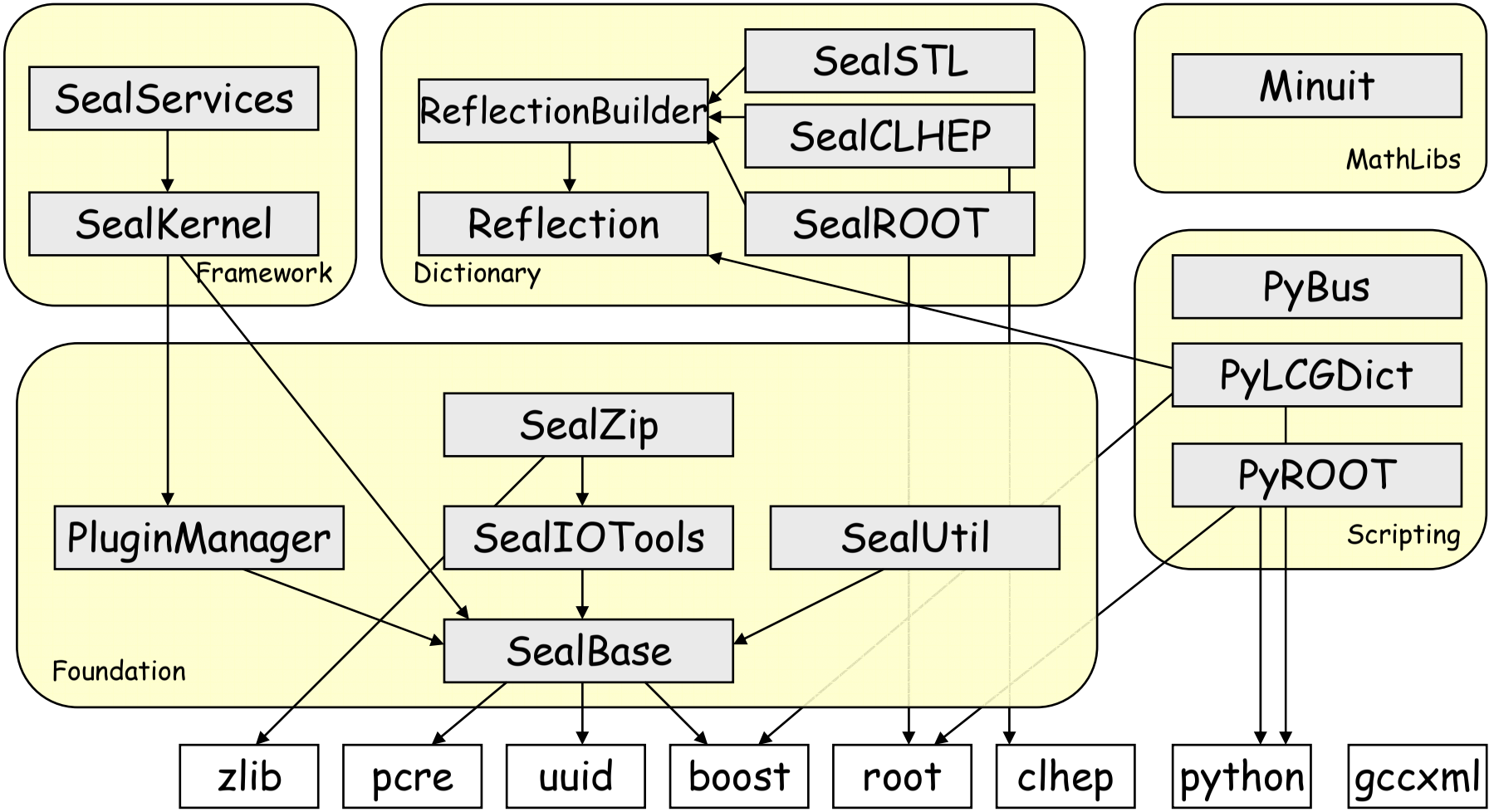
SEAL Overview

- Goals
  - Provide the software infrastructure, basic frameworks, libraries and tools that are common among the LHC experiments
  - Select, integrate, develop and support foundation and utility class libraries
  - Develop a coherent set of basic framework services to facilitate the integration of LCG and non - LCG software
- Scope
  - Foundation Class Libraries
    - » Basic types (STL, Boost, CLHEP, …), utility libraries, system isolation libraries, domain specific foundation libraries
  - Mathematical Libraries
  - Basic Framework Services
    - » Component model, reflection, plugin management, scripting services

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 (v1.3.4)



Release Road Map

Release	Date	Status	Description (goals)
V 0.1.0	14/02/03	internal	◆Establish dependency between POOL and SEAL ◆Dictionary generation from header files
V 0.2.0	31/03/03	public	◆Essential functionality sufficient for the other existing LCG projects (POOL) ◆Foundation library, system abstraction, etc. ◆Plugin management
V 0.3.0	23/05/03	internal	◆Improve functionality required by POOL ◆Basic framework base classes
V 1.0.0	18/07/03	public	◆Essential functionality sufficient to be adopted by experiments ◆Collection of basic framework services ◆Scripting support
V 1.1.0	05/09/03	public	◆Corrections and improvements of Framework
V 1.2.0	16/10/03	public	◆Support for ICC and VC++ compilers
V 1.3.0	25/11/03	public	◆Improvements in Plugin Manager ◆Consolidation Dictionary and Minuit
V 1.3.4	29/03/04	public	◆Bug fixes
V 1.4.0	15/04/04	in preparation	◆Streamline external dependencies ◆Dictionary service ◆GSL test suit

SEAL Products and their Usage

		ATLAS	Alice	CMS	LHCb	Non LHC
Foundation	SealBase+Util					
	SealZip					
	SealIOTools					
	PluginManager					
Framework	Component Model					
	Basic Services					
Dictionary	Reflection					
	Lcgdict tool					
	Specific Dictionaries					
Scripting	PyROOT					
	PyLCGDict					
	PyBus					
MathLibs	Minuit					

In use directly In use indirectly Planned use

Summary Status

- SEAL 1.3.4 is out since end March
  - 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
- Scripting based on Python
  - Boost.Python and PyLCGDict recommended to provide Python bindings

SEAL 2004 Work Plan

- Draft document has been discussed with Experiments and in the Architects Forum, PEB
  - http://seal.web.cern.ch/seal/documents/SEAL_Program_of_Work_20040302.doc
- This Work Plan is based on
  - 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
  - Wishes from the line management to define a more coherent collaboration model with the ROOT project
  - Input from the different SEAL work packages

AA Review Recommendations

- Foundation Libraries recommendations
  - Setup tutorials, user-guides and help developers → in the Plan
  - Remove unnecessary dependencies in external packages → in the Plan
  - Convergence between the SEAL and ROOT plug-in manager → not in the Plan
- Math libraries recommendations
  - Concerns about the future of Minuit, GSL and CLHEP
  - Careful testing to guarantee reliable physics results → in the Plan
  - Provision of a coherent set of libraries including dictionaries → in the Plan
- Dictionary recommendations
  - Concerns about the size of dictionaries → already done
  - Encourage unifying the dictionary with ROOT/CINT → in the Plan
- Framework recommendations
  - Discussion with the experiments to evaluate interest in framework → in the Plan
- Scripting recommendations:
  - Evaluate the interest in the interoperability between Boost and Swig before any work is committed → already done
  - Continue with the PyLCGDict automatic binding → in the Plan
  - Seek feedback from the experiment physics community in usability of python in interactive analysis → in the Plan

LCG/ROOT Collaboration Model

- Clear recommendation from internal AA review and line management to define a more coherent 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

Dictionary Plans

- The final GOAL is to have a "single" object dictionary shared between ROOT and LCG
- Development done in collaboration with ROOT team
  - Iterate the current proposal until an agreement is reached
- The steps are:
  - New Reflection API to overcome current limitations
  - Provide reference implementation of the agreed API as baseline.
  - Adapt code generation tools (gcc_xml based) to produce new dictionary descriptions.
  - Adapt existing ROOT meta-classes (TClass, TMethod,…) to the new common reflection API.
  - Develop an implementation using available CINT data structures with the goal to eliminate the need of having the LCG<->CINT gateway (POOL).

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 phase from an interactive environment, using either Python or ROOT/CINT

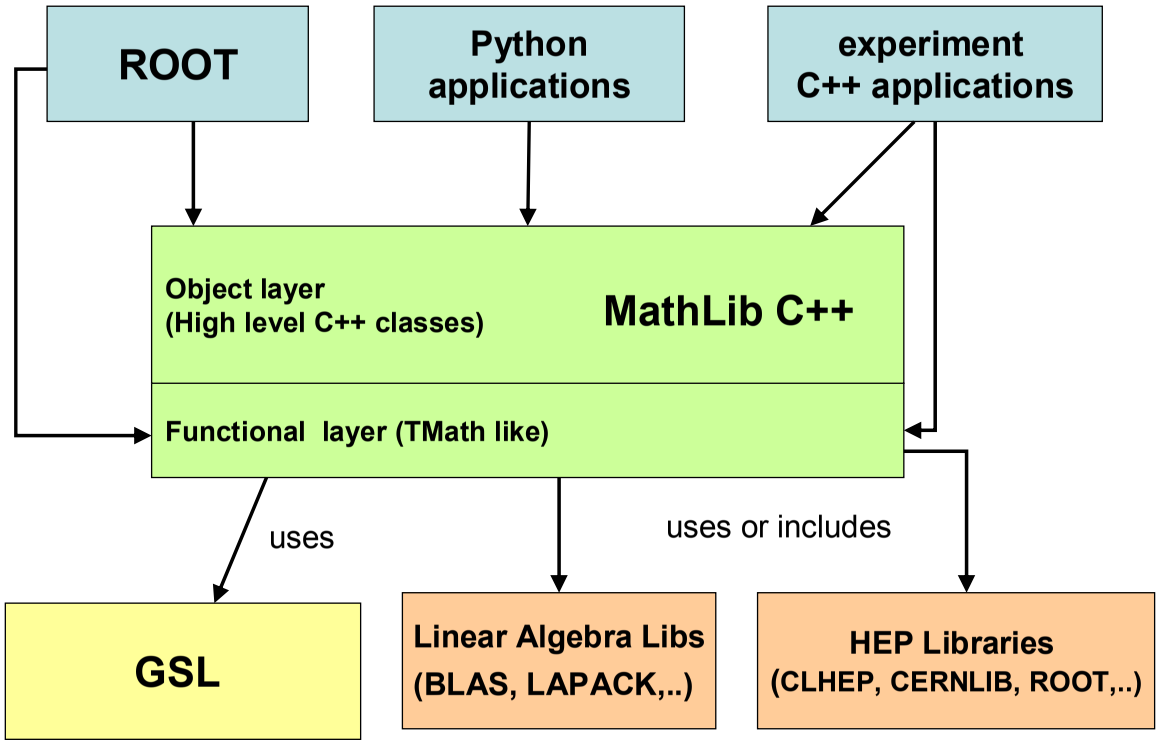
MathLibs: Work Items

- 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

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, etc.) for the development of the new MathLib C++

Summary

- Delivered a number of components that constitutes the basic foundation and utility libraries and object dictionary
- Most of the delivered components are already in use or being tested and planned to be in use by LHC experiments
- 2004 Work Plan Document available at
  http://seal.web.cern.ch/seal/documents/SEAL_Program_of_Work_20040302.doc
- 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