



# **LCG APPLICATIONS AREA PLANS FOR PHASE 2**

**Organization: CERN – LCG Project**

**Document Revision #: 1.2**

**Date of Issue: 26 September 2005**

**Editor: Pere Mato**

## Approval Signatures

Approved by: \_\_\_\_\_ Project Leader

Approved by: \_\_\_\_\_ LCG Project Leader

Prepared by: \_\_\_\_\_ Project Manager

Prepared by: \_\_\_\_\_ LCG Project Manager

## Document Change Control

This section provides control for the development and distribution of revisions to the Project Work Plan up to the point of approval.

Revision Number	Date of Issue	Author(s)	Brief Description of Change
0.1	14/06/2005	P. Mato	Initial merge of the contributions from the project leaders
0.2	16/06/2005	P. Mato	Added POOL contribution from Dirk. Added chapter "Major changes...". Modifications in ROOT chapter after discussion with Rene + resource table.
0.3	28/06/2005	P. Mato	Changes in PROOF section by Fons Updated POOL chapter by Dirk
0.4	05/07/2005	P. Mato	Updates in POOL chapter from Dirk
0.5	13/07/2005	P. Mato	Updates in POOL chapter from Dirk and summary Level milestone table
0.6	01/08/2005	P.Mato	Changes from Dirk. Added resource summary tables
0.7	31/08/2005	L. Moneta	Added SEAL work package in ROOT project section
1.0	13/09/2005	P.Mato, J.Harvey	Chapter 4 re-worked. Minor changes on milestones Extended L.A. work.
1.1	22/09/2005	P. Mato	Minor corrections from AF 22/09/05
1.2	26/09/2005	P.Mato	Feedback from Lucia Silvestris

## Table of Contents

<b>1.</b>	<b>Introduction .....</b>	<b>1</b>
<b>2.</b>	<b>Applications Area Scope and Requirements.....</b>	<b>3</b>
2.1	..High Level requirements .....	3
2.2	.. Software Architecture.....	4
2.3	.. OS Platforms.....	5
<b>3.</b>	<b>Applications Area Organization.....</b>	<b>7</b>
<b>4.</b>	<b>Major Changes in the Applications Area for Phase 2 .....</b>	<b>9</b>
4.1	.. Merger of SEAL and ROOT project teams .....	9
4.2	.. Minor redefinition of the SPI role .....	10
4.3	.. Adaptations to the scope of POOL .....	10
4.4	.. PI project to be discontinued. ....	11
4.5	.. Consolidation and change of scope in the Simulation project.....	11
<b>5.</b>	<b>Software Process and Infrastructure project (SPI) .....</b>	<b>12</b>
5.1	.. Purpose, Scope and Objectives .....	12
5.2	.. Project Organization .....	12
5.3	.. Services and Responsibilities .....	12
5.4	.. Main project milestones for 2005.....	13
<b>6.</b>	<b>Core libraries and services project (ROOT).....</b>	<b>18</b>
6.1	.. Project Organization .....	18
6.2	.. The BASE work package .....	18
6.2.1	Milestones .....	19
6.3	.. The DICTIONARY work package .....	19
6.3.1	Milestones .....	20
6.4	.. The I/O and Trees work package .....	20
6.4.1	Milestones .....	21
6.5	.. The PROOF work package.....	21
6.5.1	Milestones .....	22
6.6	.. The MATH work package .....	22
6.6.1	Milestones .....	23
6.7	.. The GUI work package .....	23
6.7.1	Ongoing work .....	24
6.8	.. The Graphics work package .....	25
6.8.1	Ongoing work .....	26

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page ii

6.8.2 Milestones .....	26
6.9 .. The GEOM work package.....	26
6.9.1 On going works.....	27
6.10 The SEAL work package .....	27
6.11 Resources .....	28
6.11.1 Staffing.....	28
<b>7. Persistency Framework Projects (POOL/COOL).....</b>	<b>30</b>
7.1 .. Purpose, Scope and Objectives .....	30
7.2 .. Project Organization .....	31
7.3 .. Persistency Framework work packages .....	32
7.3.1 Object Storage and References .....	32
7.3.2 Collections and Meta Data .....	33
7.3.3 Database Access and Distribution.....	34
7.3.4 Catalog and Grid Integration.....	35
7.3.5 Conditions Database .....	36
7.4 .. Technical process .....	37
7.5 .. Resources and Milestones.....	38
7.5.1 Staffing .....	38
<b>8. Simulation Project.....</b>	<b>40</b>
8.1 .. Purpose, Scope and Objectives .....	40
8.2 .. Project Organization .....	40
8.3 .. Simulation Framework .....	41
8.3.1 WP1 - Geometry Description Markup Language .....	41
8.3.2 WP2 - Geant4 geometry object persistency .....	41
8.3.3 WP3 - Python interfaces to Geant4 .....	42
8.3.4 WP4 – Simulation Framework for physics validation .....	42
8.3.5 WP5 - Monte Carlo truth handling .....	42
8.3.6 Sub-project Milestones .....	42
8.4 .. Physics Validation.....	43
8.4.1 WP1 – Impact of simulation on LHC physics .....	44
8.4.2 WP2 - Electromagnetic physics .....	44
8.4.3 WP3a – Hadronic physics: calorimetry .....	44
8.4.4 WP3b – Hadronic physics: inner detectors .....	45
8.4.5 WP3c – Hadronic physics: background radiation .....	45
8.4.6 WP4 – Validation of the simulation environment .....	46
8.4.7 Sub-project Milestones .....	46
8.5 .. Geant4 .....	47
8.5.1 WP1 - Geometry, Field and Transportation .....	47

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page iii

8.5.2 WP2 - Software Management.....	48
8.5.3 WP3 - EM Physics and error propagation .....	48
8.5.4 WP4 - Hadronic Physics .....	49
8.5.5 WP5 - System Testing.....	50
8.5.6 WP6 - Acceptance Suite.....	50
8.5.7 WP7 - Coordination and Releases .....	51
8.5.8 Sub-project milestones .....	51
<b>8.6 .. Fluka .....</b>	<b>52</b>
8.6.1 Summary of milestones .....	53
<b>8.7 .. Garfield.....</b>	<b>53</b>
8.7.1 WP1 – Electric field and Diffusion.....	54
8.7.2 WP2 – Gas and Transport.....	54
8.7.3 Sub-project milestones .....	54
<b>8.8 .. Generator Services .....</b>	<b>54</b>
8.8.1 WP1 - The Generator library (GENSER) .....	55
8.8.2 WP2 - Storage, Event Interfaces and Particle Services.....	55
8.8.3 WP3 - Public Event Files and Monte Carlo Database .....	55
8.8.4 WP4: Monte Carlo Validation.....	56
8.8.5 Sub-project milestones .....	56
<b>8.9 .. Technical process .....</b>	<b>57</b>
<b>8.10 Resources .....</b>	<b>57</b>
8.10.1 Staffing.....	57
<b>9. Applications Area Summary .....</b>	<b>59</b>
9.1 .. Resources .....	59
9.2 .. Milestones .....	59
9.2.1 Level-1 proposed milestones.....	59
9.2.2 Level-2 proposed milestones.....	60
<b>10. References.....</b>	<b>64</b>

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page iv	

## 1. Introduction

The Applications Area of the LCG Project is concerned with developing, deploying and maintaining that part of the physics applications software and associated supporting infrastructure software that is common among the LHC experiments.

This area is managed as a number of specific projects with well-defined policies for coordination between them and with the direct participation of the primary users of the software, the LHC experiments. It has been organized to focus on real experiment needs and special attention has been given to maintaining open information flow and decision making. The experiments set requirements and monitor progress through participation in the bodies that manage the work programme (SC2, PEB and Architects Forum). Success of the project is gauged by successful use, validation and deployment of deliverables in the software systems of the experiments. The Applications Area is responsible for building a project team among participants and collaborators; developing a work plan; designing and developing software that meets experiment requirements; assisting in integrating the software within the experiments; and providing support and maintenance.

The project started at the beginning of 2002 and recently completed the first phase in its programme of work. The scope and highlights of activities in Phase 1 are:

- The establishment of the basic environment for software development, documentation, distribution and support. This includes the provision of software development tools, documentation tools, quality control and other tools integrated into a well-defined software process. The Savannah project portal and software service has become an accepted standard both inside and outside the project. A service to provide ~100 third party software installations in the versions and platforms needed by LCG projects has also been developed.
- The development of general-purpose scientific libraries, C++ foundation libraries, and other standard libraries. A rather complete set of core functionality has already been made available in public releases by the SEAL and ROOT projects, and has been used successfully in both LCG and experiment codes.
- The development of tools for storing, managing and accessing data handled by physics applications, including calibration data, metadata describing events, event data, and analysis objects. The objective of a quickly-developed hybrid system leveraging ROOT I/O and an RDBMS was fulfilled with the development of the POOL persistency framework. POOL was successfully used in large scale production in ATLAS, CMS and LHCb data challenges in which >400 TB of data were produced.
- The adaptation and validation of common frameworks and toolkits provided by projects of broader scope than LHC, such as PYTHIA, GEANT4 and FLUKA. Geant4 is now firmly established as baseline simulation engine in successful ATLAS, CMS and LHCb production, following validation tests of physics processes and by proving to be extremely robust and stable.

The work of the Applications Area is conducted within projects. At the time there are four active projects: software process and infrastructure (SPI), persistency framework

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 1

(POOL), core software common libraries and components (ROOT), and simulation (SIMU).

The purpose, scope and objectives of each of the projects will be described in sections 4 to 7. Each project section will include the main expected deliverables for Phase II with more details and emphasis in the next 12 months. At section 8 we will summarize the resource and the main milestones for the complete Applications Area.

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 2

## 2. Applications Area Scope and Requirements

### 2.1 High Level requirements

A basic set of high level requirements were established at the start of Phase 1 of the project. Here we recall those that have guided development work so far.

It is evident that software environments and optimal technology choices evolve over time and therefore LCG software design must take account of the >10 year lifetime of the LHC. The LCG software itself must be able to evolve smoothly with it. This requirement implies others on language evolution, modularity of components, use of interfaces, maintainability and documentation. At any given time the LCG should provide a functional set of software with implementations based on products that are the current best choice.

The standard language for physics applications software in all four LHC experiments is C++. The language choice may change in the future, and some experiments support multi-language environments today. LCG software should serve C++ environments well, and also support multi-language environments and the evolution of language choices.

LCG software must operate seamlessly in a highly distributed environment, with distributed operation enabled and controlled by components employing Grid middleware. All LCG software must take account of distributed operation in its design and must use the agreed standard services for distributed operation when the software uses distributed services directly. While the software must operate seamlessly in a distributed environment, it must also be functional and easily usable in 'disconnected' local environments.

LCG software should be constructed in a modular way based on components, where a software component provides a specific function via a well-defined public interface. Components interact with other components through their interfaces. It should be possible to replace a component with a different implementation respecting the same interfaces without perturbing the rest of the system. The interaction of users and other software components with a given component is entirely through its public interface. The component architecture and interface design should be such that different implementations of a given component can be easily interchanged provided that they respect the established interfaces. Component and interface designs should not, in general, make assumptions about implementation technologies; they should be as implementation-neutral as possible.

A principal requirement of LCG software components is that they integrate well in a coherent software framework, and integrate well with experiment software and other tools. LCG software should include components and employ designs that facilitate this integration. Integration of the best of existing solutions as component implementations should be supported, in order to profit from existing tools and avoid duplication.

Already existing implementations which provide the required functionality for a given component should be evaluated and the best of them used if possible (re-use). Use of existing software should be consistent with the LCG architecture.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 3	

LCG software should be written in conformance to the language standard. Platform and OS dependencies should be confined to low level system utilities. A number of Hardware/OS/compiler combinations (platforms) will be supported for production and development work. These will be reviewed periodically to take account of market trends and usage by the wider community.

Although the Trigger and DAQ software applications are not be part of the LCG scope, it is very likely that such applications will re-use some of the core LCG components. Therefore, the LCG software must be able to operate in a real-time environment and it must be designed and developed accordingly, e.g. incorporating online requirements for time budgets and memory leak intolerance.

## 2.2 Software Architecture

Applications Area software must conform in its architecture to a coherent overall architectural vision; must make consistent use of an identified set of core tools, libraries and services; must integrate and inter-operate well with other LCG software and experiment software. This vision was established in a high level ‘blueprint’ for LCG software which provided the guidance needed for individual projects to ensure that these criteria are met [1].

LCG software is designed to be modular, with the unit of modularity being the software component. A component internally consists of a number of collaborating classes. Its public interface expresses how the component is seen and used externally. The granularity of the component breakdown should be driven by that granularity at which replacement of individual components (e.g. with a new implementation) is foreseen.

Components are grouped and classified according to the way the way in which they interact and cooperate to provide specific functionality. Each group corresponds to a domain of the overall architecture and the development of each domain is typically managed by a small group of 5-10 people. The principal software domains for LCG Applications Area software are illustrated schematically in Figure 1. Software support services (management, packaging, distribution etc.) are not shown in this figure.

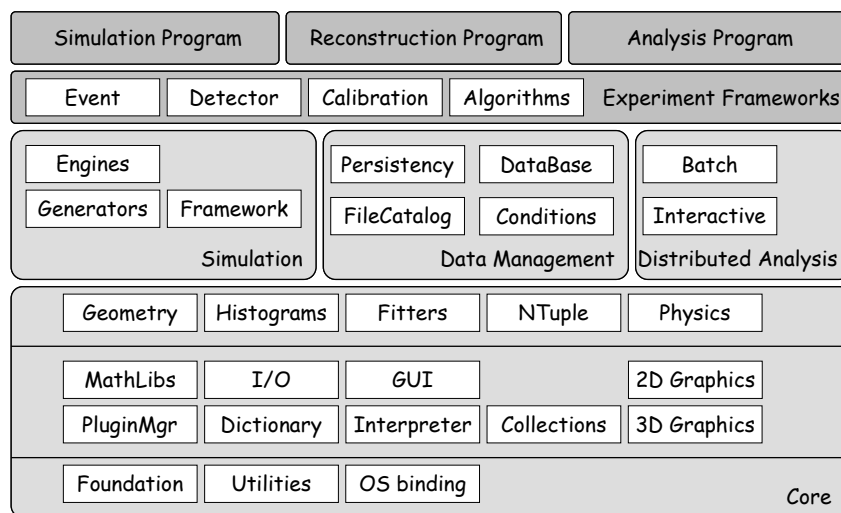


Figure 1 : Physics applications software domain decomposition

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 4	

The **Core Software Domain** provides basic functionality needed by any application. At the lowest level we identify the foundation libraries, utilities and services employed that are fairly independent class libraries (e.g. STL, or a library providing a Lorentz Vector). Above this are core services supporting the development of higher level framework components and specializations such as the plug-in manager and object dictionary by which all parts of the system have knowledge of, and access to, the objects of the system. Other core software services include command line environments for interactive and batch (script) access to the functionality of the system, as well as general graphics and GUI tools that can be used to build experiment-specific interfaces but which are not themselves experiment-specific. Histogramming, ntuples, fitting, statistical analysis, and data presentation tools also contribute to Core functionality.

Above the Core software are a number of specialized frameworks that offer services specific to particular domains. The **Data Management Domain** covers object persistency, file cataloguing, event-specific data management, and detector-conditions-specific data management. In general, the domain of expertise stays in the area of relational databases applications development. Support and LCG-directed development of simulation toolkits such as Geant4 and Fluka and ancillary services and infrastructure surrounding them are part of the **Simulation Domain**. Ancillary services surrounding event generators (e.g. standard event and particle data formats, persistency, configuration service), and support and distribution of event generator software, are also in the scope of common project activities. The **Distributed Analysis Domain** is the area where the physicist and physics application software interface to Grid middleware and services in order to support job configuration, submission and monitoring, distributed data management and Grid-enabled analysis. The scope of common activities in this area has still to be specified

Experiment applications are built on top of specialized frameworks which are specific to the experiment and not in LCG scope.

### 2.3 OS Platforms

The LHC experiments and the computer centres of universities and laboratories need to run LCG software on a large variety of platforms and operating systems, in several flavours and versions. Therefore, in order to guarantee portability, the software must be written following the most common standards in terms of programming languages and operating systems. Applications Area software is being routinely developed and run on a number of different compilers and operating systems, including Red Hat Linux, Microsoft Windows, and Apple Mac OSX, both with gcc and with their C++ proprietary compilers. This approach helps to ensure conformance to language standards and allows the project to manage dependencies on platform-specific features, both on 32-bit and 64-bit hardware architectures. Applications Area projects are involved in the certification and in the verification of new versions of compilers or operating systems at CERN.

The “production” platforms currently supported are:

- Red Hat 7.3 with gcc 3.2 and gcc 3.2.3 - the Linux reference platform for the LHC experiments and for the main computer centres. Red Hat 7.3 will be stopped by end 2005.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 5

- Scientific Linux 3 with gcc 3.2.3, and in the near future also with gcc 3.4.3 - the new Linux reference platform for CERN and other large HEP laboratories. This is binary compatible with Red Hat Enterprise 3.

In addition “development-only” platforms are supported that have better development tools and are therefore used by many programmers and users to increase productivity and assure software quality:

- Microsoft Windows, with the Visual C++ 7.1 compiler and CygWin;
- Mac OSX 10.3 with gcc 3.3, and soon 10.4 probably with gcc 4.

Any changes to the list of supported platforms or compilers is discussed and approved at the Architects Forum, where all the LHC experiments are represented. When a new platform is a candidate to become supported firstly all LCG software and external packages are re-compiled and re-built in order to assess the implications and changes needed for the new platform to become fully supported.

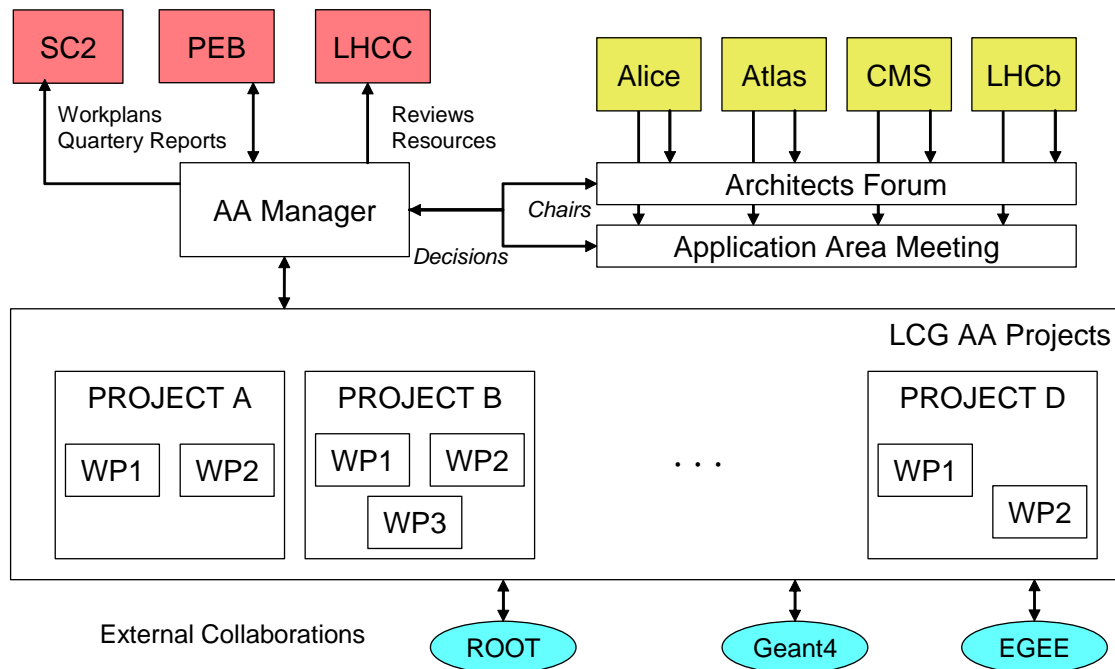
Platforms that will likely be supported in the near future are:

- SLC3 Linux on AMD 64-bit processors as an additional production platform;
- GCC 3.4.4 compiler on all Linux platforms to take advantage of better performance.
- Mac OSX 10.4 as development platform, to resolve issues related to loading of dynamic libraries.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 6	

### 3. Applications Area Organization

Applications Area work in the various activity areas described above is organized into projects, each led by a Project Leader with overall responsibility for the management, design, development and execution of the work of the project. The Applications Area Manager has overall responsibility for the work of the Applications Area.



**Figure 2: Applications Area organization**

Work in the projects must be consistent and coherent with the architecture, infrastructure, processes, support and documentation functions that are agreed application area-wide. Larger projects may in turn be divided in sub-projects and later into work packages with ~1-3 FTE activity levels per work package.

An Architects Forum (AF) consisting of the applications area leader (chair), the architects of the four LHC experiments, and other invited members provides for the formal participation of the experiments in the planning, decision making and architectural and technical direction of applications area activities. Architects represent the interests of their experiment and contribute their expertise. The AF decides the difficult issues that cannot be resolved in open forums such as the applications area meeting. The AF meets every 2 weeks or so.

The Applications Area Meeting takes place fortnightly and provides a forum for information exchange between the project and the LHC experiments.

The Applications Area work breakdown structure, milestones and deliverables for all aspects of the project are documented in this plan and are also maintained on the AA web at <http://lcgapp.cern.ch/project/mgmt>. The work breakdown structure maps directly onto the project breakdown of the Applications Area. The schedule of milestones for the

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	Version	<b>1.2</b>
				Page	<b>7</b>

completion of deliverables is similarly organized. Milestones are organized at three levels:

- Level 1: the highest level. A small, select number of very important milestones are at this level. These milestones are monitored at the LHCC level.
- Level 2: the ‘official milestones’ level. Milestones at this level chart the progress of applications area activities in a comprehensive way. Each project has a small number of milestones per quarter at this level. These milestones are monitored at the LCG project level (PEB, SC2).
- Level 3: internal milestones level. Milestones at this level are used for finer grained charting of progress for internal applications area purposes. These milestones are monitored at the AA level.

Milestones include integration and validation milestones from the experiments to track the take-up of AA software in the experiments.

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 8

## 4. Major Changes in the Applications Area for Phase 2

While AA Phase 1 has been mainly a development activity of a number of new software products, in Phase 2 we need to establish what is the desired level of long-term support and maintenance that is required for these products that are essential for the operation of the LHC experiments. To facilitate maintenance we need to minimize duplication and put in practice re-use of software and infrastructure across projects. The ultimate goal is to facilitate as much as possible the maintenance of the AA software at the end of LCG project when the resources associated to LCG will not be there.

A number of major changes in the project structure and objectives for the Applications Area for Phase 2 have been discussed among the project leaders and experiment representatives. The proposed changes have been reviewed in depth in the AA internal review that took place beginning of April [5]. The proposed changes are summarized as follows:

### 4.1 Merger of SEAL and ROOT project teams

This is probably the most important change made to the Application Area on changing from Phase 1 to Phase 2. The motivations for this merger are the following:

- to optimize use of resources by avoiding duplicate development
- to provide a coherent set of products developed and maintained by the Application Area for the benefit of its clients i.e. the LHC experiments
- to have the ROOT activity fully integrated in the LCG organization (planning, milestones, reviews, resources, etc.),
- to ease the long-term maintenance and evolution of a single set of software products.

In practice the SEAL and ROOT teams will merge into single team with a combined program of work delivering the union of the existing functionality in ROOT and SEAL.

Initially the SEAL libraries in use by the LHC experiments will continue to be maintained as long as they are still in use. The migration to new merged libraries will be done adiabatically in coordination with the experiments. End-user compatibility will be provided at each step in the process and the experiments will be involved in the definition of the new functionality and implementation details. The full migration towards a unique and coherent set of libraries is expected to take two years. The achievement of this task will be proposed as a Level-1 milestone.

The AA internal review gave a positive recommendation concerning this proposal and encouraged further discussion of the technical details of the planning in the Architects Forum (AF). A process for addressing all the topics that require detailed plans was subsequently agreed by the AF. The process is as follows:

- Produce a list of topics (functionality) that need to be merged or require attention. This list is maintained at <http://lcgapp.cern.ch/project/mgmt/coretopics2.html>
- Experiments should prioritize the list.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 9

- Each topic will be handled individually. All the topics can not be handled in parallel, but they do not need to be treated completely sequentially.
- AA projects and experiments will assign people for each of the topics and informal discussions will be organized to gather the requirements, constraints, and issues related to design and implementation.
- A written proposal for each topic will be produced after a period of 2-3 weeks specifying the agreed functionality, API, implementation details, impact for the experiments, time scale for completion, and so on.
- An open AA meeting will be organized to present and discuss each topic.
- The AF will decide and give green light for the implementation.

This process is being applied and the first decisions have been taken and are already reflected in the detailed plans for each of the projects.

More recently, the AF has agreed to keep the existing libraries of the SEAL project for a much longer period of time available for the experiments, so that they are not forced to migrate their application software. In view of this, an additional work package in the ROOT project has been created to ensure the maintenance of the SEAL/PI libraries

## 4.2 Minor redefinition of the SPI role

The SPI project should continue to provide and run a number of services of common interest (savannah project portals, external libraries service, etc.) for the LCG AA projects and for the LHC experiments. In addition SPI will continue to provide and maintain the software development infrastructure by filling a number of key roles for the Application Area , such as Software Librarian, Webmaster, etc. Some specific support activities (release management, quality assurance and documentation) will be distributed to people working in the development projects (POOL etc.). They will perform these tasks in addition to their specific development tasks for the project.

## 4.3 Adaptations to the scope of POOL

The domain of expertise of the POOL project will continue to be data persistency, data management, deployment in the Grid and relational databases. The project is currently delivering two major products: POOL and COOL. POOL is the object persistency framework that CMS, ATLAS and LHCb are basing their event data store. Part of the POOL deliverable is the relation database abstraction package (RAL), which facilitates a generic access to a number of database implementations. COOL is a database application for managing detector conditions. These two main products will continue to be developed and their deployment and integration in a grid environment will be taken care. Eventually the software packages will need to be adapted to the changes in the underlying software packages due to the merge of SEAL and ROOT.

There are no major changes proposed in the structure of the project in terms of work packages and organization.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 10	

#### 4.4 PI project to be discontinued.

The project as such will be discontinued and the existing software libraries in use will be absorbed by the client projects. In general the idea is that after an inventory of the existing libraries, the one not used will be abandoned, the ones used by one experiment or framework will be moved to the experiment or framework and the remaining ones moved to the new SEAL+ROOT project.

#### 4.5 Consolidation and change of scope in the Simulation project

The simulation project will continue to cover the domains of detector simulation, event generators, physics validation, and more software-oriented tasks related to the integration of simulation engines in application software frameworks. A new sub-project has been started to manage the further development of the Garfield tool which is widely used by the LHC experiments for the simulation of gaseous detectors. During Phase 2, the simulation projects will continue to adapt towards use of the core software and software development environment provided by the other AA projects, for example for implementing interactivity and persistency.

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 11

## 5. Software Process and Infrastructure project (SPI)

### 5.1 Purpose, Scope and Objectives

The AA software projects share a single development infrastructure; this infrastructure is provided by the SPI project. A set of basic services and support are provided for the various activities of software development. The definition of a single project managing the infrastructure for all the development projects is crucial in order to foster homogeneity and avoid duplications in the way the AA project develop and manage their software.

All of the existing LCG software projects use this software development infrastructure. The project also interacts with all LHC experiments (Alice, Atlas, CMS, and LHCb) who also make use of the services managed by SPI, in particular Savannah, external software libraries, and testing frameworks. The SPI project also gives support to CERN/IT projects.

### 5.2 Project Organization

The project has now delivered all initial versions of the foreseen tools and services and its current status can be described as being in a steady state mode of operation. Manpower is extremely tight at present, sufficient for essential operations but impacting the responsiveness of the project to new requests for service and tool enhancements and extensions. Injection of experienced FTEs would improve the responsiveness and reduce the internal stresses arising from people being spread too thinly over the required tasks.

### 5.3 Services and Responsibilities

The following table lists the current services and tasks being carried out by the SPI project together with the names of the people responsible for them.

Service and Responsibilities SPI 2005		
Service	On-going Activities and Tasks	Who
<b>Software Librarian, build and release</b>	Perform releases and pre-releases of the LCG AA software. Automate build and release functions. Work with Librarians of the AA projects and LHC experiments in order to provide what is needed by the users.	A.Pfeiffer (CERN Fellow also to be assigned in near future in addition)
<b>External Software</b>	Install and upgrade, on the LCG platforms, all external software needed by the AA projects and LHC experiments (via AF and LIM). Automation of the existing installations.	E.Poinsignon Y.Patois (to be replaced by S.Roiser and B.Bellenot at end of 2005)

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 12

<b>Service and Responsibilities SPI 2005</b>		
<b>Software Distribution</b>	Basic release management (tarfiles only for the moment) to generate installation kits for the LCG software and provide a download site.	E.Poinsignon A.Pfeiffer
	Define, develop and support a release and packaging service.	Y.Patois
<b>LCG Software Configuration</b>	Provide software configuration and build information for the software projects on all supported platforms (CMT and Scram files)	E.Poinsignon A.Pfeiffer
<b>Savannah Dev. Portal</b>	User support. Functionality enhancements. Maintenance of the service and bug fixing. Work/Merge with Savannah open source.	Y.Perrin
<b>QA and Policies Responsible</b>	Quality Assurance metrics and measurements, and verification of the policies in LCG projects. Generation of QA reports for all AA projects, concerning bug fixing and test coverage.	J.Benard (to be replaced by M.Gallas April 06)
<b>Testing Frameworks</b>	User support, maintenance, configuration, development of the tools used for testing (CppUnit, PyUnit, Oval, QMTTest).	J.Benard (to be replaced by M.Gallas April 06)
<b>Code Doc. And Web applications</b>	User support. Maintenance, configuration, development of the tools used for the code documentation (Doxygen, lxr, ViewCVS).	A.Aimar
<b>LCG policies, templates</b>	Define new policies and templates on the existing activities; foster the definition of new standards.	A.Aimar
<b>Web management, Documentation, Wiki, AOB</b>	Basic web templates for SPI and for other projects. Web definition, support and editing.	A.Aimar
	Development of a general infrastructure for a common LCG documentation and workbook.	
	Basic workbook development and documentation for internal use in SPI.	All SPI members

Table 1 List of services and tasks for SPI

## 5.4 Main project milestones for 2005

All SPI milestones are in one single list, grouped by service. Milestones are added and changed following the decisions and priorities defined by the Architects' Forum of the LCG Applications Area.

The SPI services (in Section 5.3) have their own more detailed "to do" and tasks lists, each with specific features, improvements, installations, etc. The list below includes only all project-level major milestones, not the ongoing activities.

The resources in the SPI project are sufficient for the continuation of the existing services in 2005 at current level, and also to fulfil the milestones that follow.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 13

<b>ID</b>	<b>Milestone</b>	<b>Date</b>	<b>Who</b>
<b>Build system</b>			
<i>BUIL01</i>	<i>Toolbox for scram v1 provided (manually)</i>	<i>Jun-05</i>	<i>EP, AP</i>
<i>BUIL02</i>	<i>gcc 3.4.3(4) on IA32, AMD64</i>	<i>Jun-05</i>	<i>AP, EP, YPa</i>
<i>BUIL03</i>	<i>Definition of products and partitions of the LCG software. Individual partial releases and bug releases.</i>	<i>Jul-05</i>	<i>AP, EP, YPa</i>
<i>BUIL04</i>	<i>MAC OSX 10.4 support; (gcc4.0)</i>	<i>Jul-05</i>	<i>AP, EP, YPa</i>
<i>BUIL05</i>	<i>Productize the suite of release build tools and distribute for general use</i>	<i>Jul-05</i>	<i>AP</i>
<i>BUIL06</i>	<i>Definition of build information for projects in XML (dep. on BUIL03)</i>	<i>Aug-05</i>	<i>YPa, AP, EP</i>
<i>BUIL07</i>	<i>Independent build and release of partitions of the LCG software (dep. on BUIL03)</i>	<i>Oct-05</i>	<i>YPa, AP, EP</i>
<i>BUIL08</i>	<i>Generate CMT configuration information from XML configuration (dep. on BUIL03)</i>	<i>Nov-05</i>	<i>YPa, AP</i>
<i>BUIL09</i>	<i>Generate SCRAM V1 build information from XML configuration (dep. on BUIL03)</i>	<i>Nov-05</i>	<i>YPa, AP</i>
<i>SPI01</i>	<i>CMT and SCRAM generated automatically from XML</i>	<i>Oct-05</i>	
<i>SPI02</i>	<i>Independent build and release of partitions of the LCG software</i>	<i>Nov-05</i>	
<b>Distribution</b>			
<i>DIST01</i>	<i>Pacman cache available for binaries of current releases</i>	<i>Jun-05</i>	<i>AP, EP, YPa</i>
<i>DIST02</i>	<i>Pacman cache available for sources of current releases</i>	<i>Jul-05</i>	<i>AP, EP, YPa</i>
<i>DIST03</i>	<i>Automation of generation of caches from XML dependency files</i>	<i>Sep-05</i>	<i>YPa</i>
<i>DIST04</i>	<i>Complete and document XML format</i>	<i>Aug-05</i>	<i>YPa</i>
<i>DIST05</i>	<i>Selective download for parts of GENSER</i>	<i>Aug-05</i>	<i>AP</i>
<i>DIST06</i>	<i>Productize distribution system</i>	<i>Oct-05</i>	<i>AP, EP, YPa</i>

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	Version <b>1.2</b>	Page 14	

SPI03	Pacman caches for binaries and sources	Jul-05	
SPI04	Automated distribution system in place, generate from XML description	Oct-05	
External Software			
<i>EXT01</i>	<i>Automate all packages. Write XML procedures for all packages of the latest configuration</i>	<i>Jun-05</i>	<i>EP</i>
<i>EXT02</i>	<i>Document procedures for selection of new packages and how to phase out existing packages</i>	<i>Nov-05</i>	<i>EP</i>
<i>EXT03</i>	<i>AMD64/IA32 gcc-3.4 and MAC OSX 10.4 support</i>	<i>Sep-05</i>	<i>EP, YPa, AP</i>
<i>EXT04</i>	<i>Procedure for non-AF packages: Define how other people can contribute to the installation of new packages and their maintenance (to then propose to the AF)</i>	<i>Nov-05</i>	<i>EP</i>
Savannah			
<i>SAV01</i>	<i>Provide documentation for end users and project administrators (series of how-to, Wiki, etc)</i>	<i>Aug-05</i>	<i>YPe</i>
<i>SAV02</i>	<i>Move service to SLC3 servers: Get SLC3 and all products that savannah relies on installed on development system and lcgapp1. Install Savannah code and database and make sure it works fine. Move production system to lcgapp1, test, and then release to users.</i>	<i>Jul-05</i>	<i>YPe, AP</i>
<i>SAV03</i>	<i>User Forums: Setup up a user forum for discussions and self-help (maybe a general user forum system)</i>	<i>Dec-05</i>	<i>YPe</i>
<i>SAV04</i>	<i>Export Feature: Provide mechanism to extract data from Savannah (define granularity, interface, format)</i>	<i>Oct-05</i>	<i>YPe</i>
<i>SAV05</i>	<i>Import Feature: Provide mechanism to import/migrate external data into savannah</i>	<i>Nov-05</i>	<i>YPe</i>
SPI07	Import/export features available in the Savannah system. Provide mechanism to import/migrate external data into savannah.	Nov-05	
SPI08	Savannah user forum setup, user and admin documentation available.	Dec-05	

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 15	

Quality Assurance			
QA01	QA procedures for LCG projects: Actively define procedure for the LCG projects, discussing with them and proposing to the AF the policy to follow (start from existing policies)	Sep-05	JB AA (AF meetings)
QA02	Test Coverage for non LCG projects: Define test coverage procedures and scripts for non LCG projects, refine and customize reports	Jul-05	JB
QA03	Profiling Tools: Investigate and propose a solution for profiling the LCG software (gprof, Intel Vtune, Valgrind, etc)	Oct-05	JB
QA04	LCG Metrics: Define a set of consistent metrics for the LCG projects (check with the AF, as there is the re-organization)	Nov-05	JB, AA (AF meetings)
SPI09	QA reporting on savannah and test coverage available and documented.	Oct-05	
Testing			
TST01	Usage of QMTest in SPI: Define where QMTest could be used in SPI, add actions accordingly	Dec-05	JB, all SPI
TST02	Present outside LCG: Testing frameworks interesting for the LHC experiments (AA review)	Dec-05	JB
TST03	Update all documentation and verify all testing tools on all platforms (including Windows and OSX)	Sep-05	JB
TST04	Install latest version of testing tools on AFS and update the configuration for the projects	Aug-05	JB
SPI10	Testing tools updated to the latest versions on the AFS area and available to LCG projects and outside.	Sep-05	
Documentation			
DOC01	LCG workbook: Workbook for all users of SPI infrastructure (project administrators, developers, users)	Sep-05	AA
DOC02	Doxygen and LXR documentation: Automated generation of snapshots and tagged versions	Jun-05	AA
DOC03	External packages: Generate doxygen for AIDA,	Jun-05	AA

Organisation CERN – LCG Project		Title/Subject Applications Area Plans Phase 2		Number	
Owner	Approved by	Date 26/09/2005	Version 1.2	Page 16	

	<i>CLHEP and ROOT</i>		
<i>DOC04</i>	<i>Cross-link doxygen: Link among all the generated doxygen webs, taking into account the correct configuration/toolbox</i>	<i>Jul-05</i>	<i>AA</i>
<i>DOC05</i>	<i>SPI Wiki: Sort out the structure and procedures for the workbook for SPI (internal and external). Document, track and coordinate services provided by IT division, like the build servers, CVS service, etc.</i>	<i>Aug-05</i>	<i>AA</i>
<i>SPI11</i>	<i>Doxygen and LXR automated and documented for the LCG software and for the packages (ROOT, AIDA, CLHEP)</i>	<i>Aug-05</i>	
<b>Training</b>			
<i>TRAI01</i>	<i>Define training for LCG users: Define training needs for the LCG users, push authors to make material and organize courses</i>	<i>Jun-05</i>	<i>AA</i>
<b>System administration and maintenance</b>			
<i>SYS01</i>	<i>Move services from rh73 to slc3: Install SLC3 on lcgapp1 (and lcgapp3), move/merge services from lcgapp2 (RH73) to lcgapp1 and test, then release to users.</i>	<i>July -05</i>	<i>AP, YPe</i>
<i>SYS02</i>	<i>Setup redundant server: Prepare lcgapp3 to be able to take over from lcgapp1 in case of emergency and test.</i>	<i>Sep-05</i>	<i>AP</i>
<b>Others</b>			
<i>OTH01</i>	<i>EGEE support and new plan for EGEE phase 2</i>	<i>Dec-05</i>	<i>AA, AP</i>
<i>OTH02</i>	<i>Testing and QA tools for distributed applications (EGEE request)</i>	<i>Dec-05</i>	<i>JB</i>

**Table 2 SPI level-2 and level-3 (italic) milestones**

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 17	

## 6. Core libraries and services project (ROOT)

### 6.1 Project Organization

Following the merge of the SEAL and ROOT projects (see Section 4.1), the ROOT project has been restructured into the following work packages (N.B. the person responsible for each work package is also indicated):

- BASE: (Fons Rademakers). System classes, CVS management, documentation and releases.
- DICTIONARY: (Philippe Canal/FNAL). Reflexion system, ROOT meta classes, CINT and Python interpreters.
- I/O: (Philippe Canal): Basic I/O, Trees, Tree queries
- PROOF (Fons Rademakers): The PROOF and xrootd systems.
- MATH (Lorenzo Moneta); Mathematical libraries, histograming, fitting.
- GUI (Ilka Antcheva): Graphical User Interfaces and Object Editors
- GRAPHICS (Olivier Couet): 2-D and 3-D graphics
- GEOM (Andrei Gheata/Alice) : The geometry system
- SEAL (Lorenzo Moneta): Maintenance of the SEAL packages

### 6.2 The BASE work package

This work-package is responsible for the entire base infrastructure that is used by all the other work-packages. The deliverables are:

- System and OS-dependent classes (implementations of the base class TSystem).
- The plug-in manager
- The ACLIC system, an automatic interface between the CINT interpreter and the local compilers and linkers. We would like to reimplement the library dependency tracking mechanism. Currently it relies on linking the new library against all the libraries that have been loaded. To solve many issues inherent with this approach we need to use a different approach equivalent (or identical) to autotools' .la files, which keep track of dependencies in a formatted text file.
- The configuration system in CVS and the build procedures. Once the merging with SEAL is completed, the migration to a central CVS service will be investigated.
- The development (every month) and production (twice a year) releases.
- The documentation system: web pages, reference and user guides. Make the documentation in the code compatible with Doxygen.
- QA with the test suite, the code checker and monitoring facilities.
- The mailing lists, ROOT Forum and bug reporting system.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 18	

### 6.2.1 Milestones

ID	Delivery date	Description
BAS1	2005/08/01	New updated plugin manager
BAS2	2005/07/15	A new edition of the ROOT Users Guide
BAS3	2005/07/15	The ROOT bug tracking system moved to Savannah

**Table 3 ROOT/BASE level-2 and level-3 (italic) milestones**

## 6.3 The DICTIONARY work package

Responsible for maintaining and developing the reflexion system, the ROOT meta classes, CINT and the interpreter interfaces. Masa Goto remains fully responsible for CINT and its evolution.

The current reflexion system is entirely provided by CINT. We plan on updating CINT to use the reflexion system provided by the Reflex system. The CINT/Reflex workshop, held in early May 2005, produced a plan for evaluation and implementation.

The code interpretation part of CINT is currently being reengineered by Masa Goto. The previous implementation had two different mechanisms for code interpretation, one for non-optimized code, another one using the generation of an internal byte-code for the optimized case. The new system will have only one case based on a new version of the byte-code generator. The new version of CINT will depend on Reflex and the new CINT tar file will also contain Reflex for consistency and satisfy the requirements of CINT users not using the ROOT system.

During the May workshop, it was decided to split the work in the following tasks:

- DIC1: Porting and adapting the SEAL Reflex classes to the ROOT environment (coding conventions and supported platforms) (Stefan Roiser)
- DIC2: PyRoot: improve the compliance to C++ (namespaces, templates, etc.) (Wim Lavrijsen)
- DIC3: Adapting the rootcint header file parser and code generator to produce code consistent with the Reflex API. (Stefan Roiser)
- DIC 4: Modifying the bulk of CINT to use the Reflex API (Philippe Canal and Stefan Roiser)
- DIC 5: Work on the new byte-code generator (Masa Goto).
- DIC 6: Adapting the ROOT Meta classes to the new API (???)
- DIC 7: Address the known deficiencies that have not been solved by using the new reflexion system (???)
- DIC 8: PyRoot: Adaptation to Reflex (Wim Lavrijsen)

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 19

Most of (DIC1) and (DIC2) have already been finished for the June release.

The upgrade of the code parser/generator (DIC3) should be available in September. Before all phases are completed, we will have to support the old rootcint as well as the new rootcint (probably using the existing cintex interface).

The evaluation phase for (DIC4) has been estimated to take about 3 months by Philippe. Assuming that major changes or additions will not be required in Reflex, we expect to finish (DIC4) for the December release. A progress report about this phase will be given to the ROOT workshop at the end of September.

Masa is working on (DIC5) since one year. We expect to plug-in this new algorithm (or one still in parallel) before the end of this year.

The (DIC5) (expected to take about one month), pyroot and (DIC8) steps, will start only once the steps (DIC1), (DIC3) and (DIC4) will have been completed.

In parallel, we also need to provide on-going support and try to solve any new issues.

### 6.3.1 Milestones

ID	Delivery date	Description
DIC1	2005/06/30	Reflex and Cintex in the ROOT v5 development release
<i>DIC2</i>	<i>2005/09/30</i>	<i>rootcint header file parser and code generator adapted to produce code consistent with the Reflex API.</i>
<i>DIC3</i>	<i>2005/09/30</i>	<i>Assessment of the work to modify CINT to Reflex.</i>
DIC4	2005/12/31	CINT fully converted to Reflex
DIC5	2006/03/31	PyRoot exclusively based on Reflex

Table 4 ROOT/DICT level-2 and level-3 (italic) milestones

## 6.4 The I/O and Trees work package

Responsible for maintaining and developing the core I/O system, TTrees and TTree queries.

Following the merge, a lot of work has already been done in the object persistency mechanism. This is already included in ROOT version 4.04 end of April.

The first development release of ROOT version 5 will include new optimisations in speed and support for additional C++ constructs. A corresponding optimisation has already been done in the Trees, in particular when using STL collections.

The main developments with Trees are

- IO1: Support for friend trees when trees are grouped into TChains.
- IO2: Support for bitmap indices (work in collaboration with LBL)
- IO3: Adaptation to PROOF
- IO4: Speed-up of the Tree query mechanism.

Organisation CERN – LCG Project		Title/Subject Applications Area Plans Phase 2		Number	
Owner	Approved by	Date 26/09/2005	Version 1.2	Page 20	

- IO5: Creation a interface between TTree and SQL databases

The main development related to TTree queries

- TRE1: Production release of the MakeProxy mechanism
- TRE2: Implementation of the auto-dereferencing of TRefs and Pool::Ref
- TRE3: Optimize TTreeFormula run-time by using the new TFormula optimization
- TRE4: In addition there are a few outstanding issues (support virtual, private, and protected inheritances, private and protected typedefs) that need to be addressed

#### 6.4.1 Milestones

### 6.5 The PROOF work package

A major upgrade of the PROOF system has been started in January. PROOF is evolving from a system that processes interactive short blocking queries to a system that also supports long running queries in a stateless client mode. It will be possible to reconnect from the same or another client session and to browse the set of running queries or retrieve the results of finished ones.

To be able to support these new features we plan to reuse the xrootd [ref] file server infrastructure and extend it with a protocol handler for PROOF. Xrootd offers an advanced, very efficient, multi-threaded connection and poller infrastructure and a protocol processor that is easily extensible. An early prototype has already shown the feasibility of this approach.

We are also working on a set of authentication plugins for xrootd that will support the most popular authentication methods: Kerberos, Globus, SRP, ssh, uid/pwd and uid/gid. By reusing xrootd as PROOF front-end daemon PROOF will directly benefit from this work.

Further work is ongoing in making the PROOF cluster setup "zero-config" by using the Apple Bonjour protocol for service discovery. By using Bonjour the PROOF server nodes will be able to discover the PROOF-config service and register themselves and their capabilities. When a user connects to the PROOF master the master will discover and use the PROOF-config service to find the slave nodes.

In addition a lot of work is being done in the area of usability and user friendliness of PROOF. Integration in the ROOT browser will allow the browsing of the ongoing PROOF sessions and the analysis products created in these session. Also all chain, friend, eventlist and index operations available in standalone ROOT sessions will be supported in PROOF.

To run PROOF on the Grid we are working on interfaces to Grid middleware that will allow us to access file catalogs, job schedulers and storage elements from ROOT and PROOF. Via these interface we can find the desired file sets in the file catalog, the

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 21	

storage elements containing the files and partition the query in such a way that slaves are started on the compute elements close to the data.

Except for the Grid extensions we expect all the milestones to be achieved before the ROOT 2005 workshop. The Grid extensions are dependent on the Grid middleware currently under development in LCG/EEGE and the different LHC experiments and the milestones are therefore more difficult to define.

Once the new system has been demonstrated further developments will largely be driven by user feedback.

### 6.5.1 Milestones

<b>ID</b>	<b>Delivery date</b>	<b>Description</b>
<i>PRF1</i>	<i>2005/07/15</i>	<i>Finish xrootd authentication modules</i>
<i>PRF2</i>	<i>2005/07/15</i>	<i>Add support for friends, eventlists and indices</i>
<i>PRF3</i>	<i>2005/08/01</i>	<i>Switch to new xrootd based front-ends, still blocking</i>
<i>PRF4</i>	<i>2005/08/15</i>	<i>Add zero-config features</i>
<i>PRF5</i>	<i>2005/09/01</i>	<i>First version of non-blocking queries</i>
<i>PRF6</i>	<i>2005/09/01</i>	<i>Add session and result browsing</i>
<i>PRF7</i>	<i>2005/09/30</i>	<i>Demonstration of new PROOF system at ROOT 2005</i>

**Table 5 PROOF level-2 and level-3 (italic) milestones**

## 6.6 The MATH work package

This work package is responsible for:

- MAT1: MathCore: a small set of mathematical functions, utilities, physics vectors and random number generators. MathCore is a self-consistent set of classes that will also be released as an independent library. A first call for comments for the vector package has been done. We expect to include a first version of this library in the ROOT development release end of June with the physics vector package. We will later migrate in the library some of the numerical algorithms currently existing in ROOT, in other classes like TF1. For the random numbers we will evaluate a prototype package compliant with the proposed design to the C++ standard and eventually re-implement the current ROOT TRandom classes using this new package.
- MAT2: MathMore: a larger set of utilities and functions. This first implementation will consist of C++ wrappers calling functions automatically extracted from GSL. A first release including the C++ wrapper developed in SEAL for special and statistical functions some numerical algorithms like adaptive integration will be available for the August release.
- MAT3: the histogramming classes. No major developments are foreseen in the near future.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 22	

- MAT4: Minimisation and Fitting classes. The current abstract interface (TVirtualfitter) will be modified and extended to support the new Linear and Robust fitters and the new version of Minuit. A new proposal with redesign of the classes is expected for the end of September and the implementations for the end of the year.
- MAT5: The new version of C++ Minuit from SEAL requires some changes and simplification in the user interface (API). The Minimization package developed by the Fermilab computing group will be evaluated for a possible merge with the new Minuit.
- MAT6: The RooFit package from BaBar is currently being introduced in CVS. Some more work to understand its impact on the other fitters and to have a complete and coherent integration with the other ROOT Math packages will have to be done later.
- MAT7: Continue a detailed evaluation of the ROOT Linear Algebra package with performance tests in experiment applications such as track reconstruction, comparing with currently existing packages.  
Evaluate also the possibility to move to template classes on the scalar type (float, double) such that the number of classes to be maintained is reduced.  
In addition, the LHC experiments would like to evaluate the impact of removing the current dependency of the L.A. classes to the ROOT framework classes, such as TObject, while keeping the existing classes fully functional. It was agreed to develop a prototype of such a library with the help of experiment effort to perform this evaluation.

### 6.6.1 Milestones

ID	Delivery date	Description
MAT1	2005/06/30	First version of new MathCore library
MAT2	2005/08/31	First version of the MathMore and random number generator library
MAT3	2005/09/30	New C++ Minuit in be released in ROOT
MAT4	2005/12/31	Finalization of the Fitting and Minimization interfaces
MAT5	2005/12/31	RootFit fully integrated (in CVS by July)
MAT6	2005/12/31	Evaluation of a prototype linear algebra package independent of TObject

Table 6 ROOT/MATH level-2 and level-3 (italic) milestones

## 6.7 The GUI work package

The Graphical User Interface sub-system of ROOT provides the following sub-systems:

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 23

- GUI1: Several implementations of the abstract interface TVirtualX used by the GUI and the 2-D graphics classes
  - o TGX11: The implementation used on Unix systems and based on X11.
  - o TGWin32: The corresponding implementation for Windows using directly the native win32 interface.
  - o TGQt: an implementation based on the Qt package from TrollTech and developed in collaboration with Valeri Fine from BNL.

These 3 implementations provide the same look & feel and behaviour on all systems.

The Qt interface supports several modes:

- o one mode where ROOT manages the graphics event loop. Qt events are forwarded to Qt.
- o one mode where Qt manages the graphics event loop. All non-Qt events are forwarded to ROOT.

In both modes a ROOT graphics widget or a Qt native widget can be included in either a ROOT managed canvas or a Qt managed canvas.

The Qt interface is currently under validation testing and is available only on Unix systems.

- GUI2: A set of high level widgets like:
  - o the canvas manager
  - o the ROOT object browser that understands a complex hierarchy of C++ objects or files.
  - o the Tree Viewer specialized to process interactive queries to Trees or Chains. It works also in the context of PROOF.
- GUI3: The object editors. The system provides a protocol to automatically create interfaces for user-defined objects and also for the most currently used ROOT objects like histograms, graphs or basic graphics primitives.
- GUI4: A GUI designer and code generator. The designer is a toolkit helping the interactive creation of a GUI. Once a new widget has been created, the system can automatically generate the corresponding C++ code, such than an iterative process calling the generated code via the interpreter (or compiled code via ACLIC) and using the drag and drop features of the editor speeds-up the building of the UI.

### 6.7.1 Ongoing work

For GUI1 is ongoing consolidation of the various interfaces. The Qt interface still requires work to become stable and it must be made available under Windows. We expect to have a more stable interface with the July/August development release. The port on Windows should be available next year (preliminary planned for the release in December).

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 24	

For GUI2, the ROOT object browser needs several upgrades to support the control of attributes in the nodes being visualized. This development is important when browsing detector geometries. Ideally we would like to have a solution this year, but with the departure of the developer, it is not clear today who can implement these new features. Improvements of TTreeView planned for the release in December.

For GUI3, the new editors are in the pipeline. A Style Editor will be implemented by a summer student. We expect to implement a Fit editor before March 2006.

For the GUI designer, originally implemented by V.Onuchin, all complex widgets still remain to be implemented in the toolkit. They will be implemented gradually as time permit. An important feature is the possibility to generate automatically the C++ code for the signal/slots.

## 6.8 The Graphics work package

This work-package is responsible for:

- GRA1: basic graphics like
  - o lines, text, polygons, markers drawing
  - o graphs to draw a set of points with or without errors (all dimensions)
  - o histograms visualisation (all dimensions)
- GRA2: generation of the various output formats like ps, eps, pdf, gif, jpg, bmp
- GRA3: image processing classes. This work is done in collaboration with Sasha Vasko, the main developer of the open source system libAfterImage. The image processing classes are becoming more and more interesting in many aspects of visualisation. They provide more functionality and are more efficient when viewing large matrices of points. This set of classes is also used to generate output formats like gif, jpg when running in batch mode. This is required by automatic tools using ROOT to produce reports that can be visualized with external viewers on the web.
- GRA4: 3-D graphics including basic 3-d primitives and detector geometries. An abstract interface to a 3-D viewer has been defined (TVirtualViewer3D) and implemented with the old X3D viewer (Unix only) and the GL viewers. A big effort is currently going on to develop a powerful GL viewer that could be used to visualize complex geometries and physics events taking advantage of the modern 3-d graphics cards available on most laptops. The new GL viewer is designed such that it communicates with the client with only a basic set of classes (TVirtualViewer3D and TBuffer3D). The viewer is independent of the original producer (e.g. the TGeo classes, or g3d shapes). This viewer is providing much more functionality than legacy GL viewers like OpenInventor or Coin3D. It is developed in collaboration with Timur Potchepstov from JINR Dubna. The following features are currently under implementation:

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 25

- automatic reloading of the GL data base from the original detector geometry data base in real time. This is a major achievement compared to classical viewers that can only show static extracts from the geometry data base.
  - slicing of the detector with cuts with 1,2 or 3 planes.
  - drawing of the outline of the shapes (like in CAD systems).
  - support for composite shapes.
  - optimisation of the GL data base with clever caching to speed-up the viewing pipeline when rotating or zooming the picture.
- GRA5: Implement a version of TPad and TCanvas based on OpenGL to allow mixing of high level 2d and 3d graphics in the same output.

### 6.8.1 Ongoing work

The usual consolidation work for GRA1 is ongoing. We intend to adapt the viewing of histograms with the 3-D options (lego, surfaces) to the new GL viewer. This will happen gradually between September 2005 and June 2006.

For GRA2, we intend to remove several restrictions in the output formats, e.g. transparent pads) and implement support for new graphics patterns in formats like pdf.

For GRA3, we will have to follow the developments in libAfterImage.

For GRA4, we intend to demonstrate the new GL viewer at the ROOT workshop in September with possible applications in Event display packages. The viewing of dynamic particles will follow and could be available for the December release.

For GRA5, it is expected to be available for the ROOT workshop in September.

### 6.8.2 Milestones

ID	Delivery date	Description
GRA1	2005/09/30	Demonstrate the new GL viewer at the ROOT workshop.

Table 7 ROOT/GRAPH level-2 and level-3 (*italic*) milestones

## 6.9 The GEOM work package

The ROOT geometry classes (TGeo) have been developed with and mainly Andrei and Mihaela Gheata from Alice. These classes support:

- description of complex detectors. All LHC detector geometries are currently supported as well as the quasi totality of existing detectors.
- automatic and efficient detection of volume overlaps or extrusions.
- navigation in these geometries with efficient computation of distances to boundaries, normals to surfaces.
- powerful visualisation package.

Organisation CERN – LCG Project		Title/Subject Applications Area Plans Phase 2		Number	
Owner	Approved by	Date	26/09/2005	Version	1.2   Page 26

- These classes have been used in conjunction with the VMC (see below) and have interfaces with Geant3, Geant4 and Fluka.

The interesting feature of the TGeo classes is that they can be used for detector simulation, reconstruction, event displays, online, etc.

### 6.9.1 *On going works*

A continuous optimisation process is going on to:

- optimize the performance
- run quality assurance tests comparing the results with other geometry
- data bases like Geant3, Geant4 and Fluka.
- add additional functions required by reconstruction programs like the concept of logical assemblies, extrapolation to a virtual surface.

This work is being done mainly by Alice in the context of their MonteCarlo validation process. The TGeo classes have been so far certified with Geant3 and Fluka.

This continuous optimisation process is expected to continue in the coming months. A major worry is the continuation of this process once the main authors contracts will finish in the middle of 2006.

## 6.10 The SEAL work package

This work package is responsible for maintaining the SEAL software which is not going to be integrated in ROOT but it is requested by the LHC experiments and other LCG projects such as POOL. No new developments for these packages are foreseen and the maintenance will be done only for the official platforms agreed by the LCG application area.

The packages of SEAL that will be maintained in the long term and they will not be part of ROOT, are the following:

- Foundation:
  - SealBase, SealIOTools, SealUtil, SealZip and PluginManager
- Framework:
  - SealKernel and SealServices

The current existing Dictionary Reflex and Cintex packages and the MathLib packages (Minuit and MathCore) will be maintained as long as the integration inside ROOT is not yet completed. In addition, they will only be discontinued in their current form with the explicit agreement of the LHC experiments.

The Dictionary Reflection and ReflectionBuilder packages and the Scripting packages PyLCGDict and PyLCGDict2, which are based on Reflection, will be maintained until Reflex will be in used in production by the experiments.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 27	

The current version of SEAL provides in addition the dictionary libraries generated using both Reflex and Reflection for some common used external packages like STL, CLHEP and ROOT. It is foreseen that the dictionary for STL and ROOT, will be released as part of ROOT. The dictionary for CLHEP will be maintained as long as the experiments will have not completed the migration from CLHEP to the new ROOT Math classes.

The SPI group will have the responsibility of building the releases on all the supported platforms.

Milestones:

ID	Delivery date	Description
SEAL1	2005/11/30	Remove from new SEAL releases Reflex and Cintex packages. Bug fix releases will continue to include them.
SEAL2	2005/11/30	Remove from new SEAL releases Minuit and MathCore. Bug fix releases will continue to include them.
SEAL3	2005/12/31	Remove from new SEAL releases Reflection, ReflectionBuilder and the Scripting packages (PyLCGDict and PyLCGDict2)

**Table 8 ROOT/SEAL level-2 and level-3 (italic) milestones**

## 6.11 Resources

### 6.11.1 Staffing

BASE : Fons (0.2) , Ilka (0.3), Jose (0.2), Philippe (0.1), Bertrand (0.1), Axel (0.2)

DICT: Stefan (0.6), Markus (0.1), Wim (0.2), Philippe (0.4), Masa (0.2)

I/O: Markus (0.2), Philippe (0.5)

PROOF: Fons (0.3), Marek (0.7), Gerri (1.0), Guenter (0.4), Maarten (0.2)

MATH: Lorenzo (0.9), Andras (0.75), Anna (0.5), Eddy (0.1)

GUI: Ilka (0.7), Valeriy (0.5), Fons (0.1), Valeri (0.1), Bertrand (0.2)

GRAF: Olivier (1.0), Richard (1.0), Andrei (0.1), Bruno (0.1), Timur (0.8)

GEOM: Andrei (0.6), Mihaela (0.2)

SEAL: Lorenzo (0.1), Stefan (0.1)

	CERN+LCG	LHC Exp.	Other	Total
MGR	1.0			1.0
BASE	0.5	0.2	0.4	1.1
DICT	0.6	0.3	0.6	1.5
I/O			0.5	0.7

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 28	

PROOF	2.0	0.4	0.3	2.6
MATH	2.15		0.1	2.7
GRAF	2.0	0.2	0.8	3.0
GUI	1.3		0.3	1.6
GEOM		0.8		0.8
SEAL	0.2			0.2
Total	9.75	2.1	2.9	14.75

**Table 9 Manpower currently available for the ROOT project by work package and funding source.**

## 7. Persistency Framework Projects (POOL/COOL)

### 7.1 Purpose, Scope and Objectives

The persistency framework project provides a general persistency store for the LHC experiments to store event and associated meta data. The project has following the proposal of the LCG Persistency RTAG [6] adopted a hybrid approach for its implementation.

The LCG persistency framework consists of two main packages: POOL and COOL). The POOL project represents a generic object store which provides independent of the backend technology a consistent interface to store C++ objects into any of the POOL implementation technologies (ROOT or RDBMS). The main concept used to store and retrieve objects is the POOL token, which is generated when an object is stored. This token fully specifies the location of an object in a distributed store consisting of many files and databases. The token can later be used to quickly retrieve any object from the store. In addition, the POOL object token can be persistently stored inside other objects to represent object associations in a similar way as C++ pointers allow this for transient data. POOL has provided the main required functionality for file based object storage integrated with the grid cataloguing services. These core capabilities have been validated in several larger experiment data challenges up to the scale of 400TB. More recently POOL has provided a general relational database abstraction package (RAL), which has been successively replaced all direct (MySQL specific) database uses through out POOL and has been picked up also by experiments to implement their database applications. This package is expected to play a central role in the integration of physics database application with the distributed database environment as prepared by the LCG 3D project.

As one of the central LCG data manage packages, POOL is responsible for integrating the experiment offline frameworks with other core packages produced by the application area such as SEAL and ROOT. At the same time POOL components connect the experiment applications with the basic grid services such as file catalogs and security and distributed database services. For this reason POOL will during LCG Phase 2 need to evolve in close collaboration with both application area and grid middleware/deployment areas.

The COOL subproject provides a more specialised software package to manage time series of user data such as detector conditions or calibration constants. The package provides a high level API, which allows the user to store arbitrary payload data (relational tuples or references to complex object) together with the time interval of its validity (IOV). COOL supports to organise a large number of different data types in a hierarchical folder structure, similar as file systems allow to organise files. COOL also supports optionally to maintain for a given quantity and validity time to version the payload data. This versioning is required to represent e.g. different results for calibration constants that have been obtained from different algorithms. For conditions data time of validity, conditions folder and the optional processing version uniquely define each payload data item. The COOL implementation is build on top of the relational database abstraction (RAL) and will directly profit from its integration with grid aware database

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 30

services. COOL will allow using the POOL C++ object storage components to store more complex object payloads in databases or files. This choice will be left to the COOL users/experiments.

COOL is a more recent development that still needs to be fully integrated in the experiment frameworks and to be validated in larger scale production activities. In particular, the distributed deployment of COOL as one of the first distributed database applications will need significant preparations and tests during Phase 2 of LCG.

## 7.2 Project Organization

The persistency project is structured into work packages, which closely follow the domain decomposition of the persistency framework. Each work package is coordinated by a work package leader and is responsible for the definition of its components interfaces in collaboration with the main clients. These clients may be other POOL components or end-users. Each work package is also responsible to implement if necessary several different technology plugins for each component, to match the required operational environments or to follow technology evolution with time.

Work Package	Coordination
Catalog and Grid Integration	Zhen Xie
Storage Manager and Object References	Markus Frank
Collections and Meta Data	David Malon
Infrastructure and Testing	Giacomo Govi
Database Access & Distribution	Ioannis Papadopoulos
Conditions Database	Andrea Valassi

Roles	
Project Leader	Dirk Duellmann
Release Management	Ioannis Papadopoulos (POOL) Andrea Valassi (COOL)

POOL Experiment Contacts	Link Person
ATLAS	Ioannis Papadopoulos
CMS	Giacomo Govi / Zhen Xie
LHCb	Markus Frank

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 31

## 7.3 Persistency Framework work packages

### 7.3.1 *Object Storage and References*

This work package defines a technology independent API to store and retrieve arbitrary user objects. This includes the definition of object cross references (tokens) and the infrastructure to use them transparently from C++ (Refs). The API also defines the semantic for consistent transactions across file and database storage implementations. It provides implementations of this object store and an associated client side object cache based on ROOT I/O and an object-relational mapping component.

The main milestone in this area include the application area wide migration to the new LCG reflection package “reflex” which will affect the most POOL storage components. The implementation dependent storage managers for ROOT I/O and the relational object mapping will be upgraded to make use of the new package as well as the POOL references, which use the reflection to implement safe type casting operations. The ROOT based storage manager will take advantage of the Cintex package.

In parallel to the move to reflex POOL will provide a set of command line tools and APIs, which allows storing the object-mappings based on user provided for pre-existing relational data. These tools will support the experiment requirement to make eg relation data written directly into a supported back-end database available in a consistent way with other POOL stored objects.

After an initial Reflex based release of POOL we expect a validation phase in which the experiments evaluate the new dictionary implementation in particular with respect to compatibility with data written with older POOL releases. After successful validation POOL will release a reflex based production version and will freeze developments based on the older LCG dictionary.

Anther important improvement for the POOL deployment will be the introduction of schema loading on demand based on the SEAL dictionary service. This task will likely require work on the ROOT side to allow for the necessary callouts to inform POOL of the attempt to use a C++ class for which no dictionary has been loaded so far.

With the integration of Reflex, Cintex and on demand schema loading it will be possible to provide a seamless integration of POOL into ROOT as a analysis shell. Some clarification from the experiments about their detailed requirements (POOL plugin for ROOT or direct interpretation of POOL tokens inside ROOT) needs to be obtained to schedule the necessary developments on the POOL and ROOT side to allow transparent use of all POOL features from ROOT based analysis programs. A proposal will be produced by the project and presented to the experiments for discussion.

To allow for consistent and secure access of file based and database based data POOL will need to pickup the security infrastructure as defined by ROOT. This may involve passing LCG certificate information to a ROOT I/O version which supports SRM based files and to RAL based components for secure database access.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 32	

Further discussions will be required for the integration with xrootd as file server backend, which may have an impact on the POOL file catalog and security support. This activity will need more consultation with the experiments planning to use xrootd to come up with a common work plan proposal by the POOL and ROOT teams.

After the release of the repackaged RAL component POOL will upgrade the object relational mapping layer to the new version of this package. This will provide database lookup and client side database diagnostics also to applications storing complex C++ objects via POOL in relational databases.

### Milestones:

<b>ID</b>	<b>Delivery Date</b>	<b>Description</b>
<i>OSM01</i>	<i>2005/07/31</i>	<i>Validation release based on Reflex</i>
POOL01	2005/09/30	Reflex production release (Reflex validation by experiments completed)
<i>OSM03</i>	<i>2005/08/31</i>	<i>Command line tools to access pre-existing relational data</i>
<i>OSM04</i>	<i>2005/09/30</i>	<i>Support mapping of STL containers to BLOBs</i>
<i>OSM05</i>	<i>2005/09/30</i>	<i>On demand schema loading</i>
<i>OSM06</i>	<i>2005/10/31</i>	<i>Proposal for POOL refs and collections accessible from ROOT as analysis shell</i>
<i>OSM07</i>	<i>2005/11/30</i>	<i>Certificate based access to files and databases</i>
<i>OSM08</i>	<i>2005/09/30</i>	<i>Completed migration of Relational Storage Manager to the new RAL release</i>
POOL02	2006/03/31	Overall performance study and validation with expected requirements

**Table 10 POOL Storage Manager level-2 and level-3 (italic) milestones**

### 7.3.2 Collections and Meta Data

The collection work package provides a technology independent API to access and manage large scale collections of persistent objects. Both the collections and the objects themselves may reside in a POOL file or database. The collection API forms the developer level tools to construct queryable event collections as used by experiment analysis and book keeping activities. The API is accompanied by a set of command line tools, which allow maintaining collections similar to the administration tools, provided for the file and database catalogs. The collection implementation is increasingly integrated with the POOL object storage services. It is therefore proposed to merge this work package with the Object Storage work package.

After the new RAL release the database based collections will make use of the security, distribution and monitoring capabilities introduced there. All collection implementations will undergo a validation against the experiment requirements. For the database based

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 33

collections this activity will also be used to define the database service requirements for the associated back-end services.

### Milestones:

ID	Delivery Date	Description
<i>COL01</i>	<i>2005/09/30</i>	<i>Collection implementation based on new RAL release</i>
<i>COL02</i>	<i>2005/09/30</i>	<i>Tier 0 Collection scalability test</i>
<i>COL03</i>	<i>2005/10/31</i>	<i>Relational collection distribution tests</i>

**Table 11 POOL Collections level-2 and level-3 (italic) milestones**

### 7.3.3 Database Access and Distribution

The database access and distribution work package defines a vendor independent high level, relational database API (RAL) that is used as foundation for relational POOL components but also has been picked up for relational database applications developed by the experiments. RAL implementations for ORACLE, MySQL and SQLite are provided as part of POOL. Other back-ends may follow later as required by the experiment deployment models. POOL RAL after the first production releases went through a component review and several interface changes and functionality extensions have been requested. In particular the infrastructure for logical database service lookups (instead of hard-coded physical connect strings), consistent error and retry handling in case of connection problems and client side query summaries (top queries and query latencies) will be required to support the distributed deployment of database application.

To allow the use of the database foundation layer without introducing unnecessary dependencies on higher-level POOL components (eg for online use) the software will be packaged such that it can be used also outside of the full POOL release.

These requests will be addressed together as part of a first re-factoring release of RAL aligned with the database distribution tests in the LCG 3D project.

An important development that has been agreed recently is the development of a plug-in which will connect the POOL database access layer to data caching infrastructure such as the FroNtier system. This will allow to access read-only data via the http protocol instead of direct connections to a database server. This promises to achieve increased scalability and availability by replication of database data using standard web cache server implementations like squid. A proof-of-concept implementation of this plug-in will be provided and tested as part of the LCG 3D work plan against application workloads provided by the experiments. This database caching approach is expected to significantly simplify the deployment of replicated relational data to higher LCG tiers for those database applications for which the cache consistency issues can be controlled. Introducing this cache in the POOL database abstraction layer will make this new access method available to all higher-level components and application eg Conditions data stored via COOL.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 34

**Milestones:**

<b>ID</b>	<b>Delivery Date</b>	<b>Description</b>
<i>RAL01</i>	<i>2005/04/30</i>	<i>RAL component review</i>
<i>RAL02</i>	<i>2005/08/31</i>	<i>Complete support for database bulk operations (adding bulk update and delete required for COOL)</i>
<i>RAL03</i>	<i>2005/08/31</i>	<i>RAL validation release (implementing the RAL review outcome and separate packaging)</i>
<i>RAL04</i>	<i>2005/09/30</i>	<i>Proof-of-concept plug-in for proxy-cache database access</i>
<i>RAL05</i>	<i>2005/10/31</i>	<i>RAL production release</i>
<i>RAL06</i>	<i>2005/11/30</i>	<i>Production release of proxy-cache access plugin</i>
<i>RAL07</i>	<i>2005/12/31</i>	<i>Native MySQL plugin (instead of ODBC based implementation)</i>

**Table 12 POOL/RAL level-2 and level-3 (italic) milestones****7.3.4 Catalog and Grid Integration**

The catalog component is responsible for the management of consistent sets of POOL files and databases associated meta data. For grid disconnected use cases native POOL implementations exist (based on XML or database access via RAL). For deployment in grid environments POOL provides adaptors that connect POOL to the catalog services provided by the grid middleware (eg LFC, fireman, GLOBUS RLS). To achieve this POOL has with release V2.1 separated the catalog lookup interfaces from a now optional meta data interfaces. This has allowed the use also of those grid catalog services which do not support the definition of catalog meta data support. This separation will also simplify a possible direct integration with experiment provided meta data systems. After the new file catalog adaptors have been evaluated by the experiments POOL will release production versions, which take into account the experiment feedback, and bug fixes from the middleware developers. At this point we would like to de-support the older EDG-RLS implementations, for which the development has already been stopped in agreement with the experiments.

After the implementation of the review comments to the POOL/RAL package also the POOL relational catalog will be upgraded to make use of the new version of this package and this catalog (with MySQL, Oracle and SQLight as backend) will become production implementation for local/non-grid use. It is proposed to de-support the older direct implementation against MySQL at this point.

**Milestones**

<b>ID</b>	<b>Delivery Date</b>	<b>Description</b>
<i>CAT01</i>	<i>30/06/2005</i>	<i>Separate lookup and meta data interfaces</i>

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 35

CAT02	31/08/2005	Production release of LCG catalog adaptors (level 2)
CAT03	30/09/2005	<i>Relational catalog upgrade to new RAL (including database lookup and security)</i>

**Table 13 POOL Catalog level-2 and level-3 (italic) milestones**

### 7.3.5 Conditions Database

The Conditions Database subproject is responsible for providing the software and tools to store, manage and distribute conditions data (detector data that vary with time and may exist in several versions, such as calibration and alignment data). Since Q4 2004, the subproject has focused on the development of COOL, a new software implementation designed to merge the two pre-existing Oracle and MySQL based packages and extend their functionalities using a unified approach. COOL has been designed to avoid duplication of efforts by making maximum use of the software and tools provided by the SEAL and POOL projects. COOL provides an abstract interface to store and retrieve conditions data using the same user code to access different back-ends. Presently, only one relational implementation of this API exists, providing access to Oracle and MySQL relational databases via the POOL RAL component.

The focus on the COOL development side will stay on the completion of the functionality and performance needed by the experiments for their first deployment of COOL at CERN in 2005. A prioritised list of requirements has been gathered based on experiment input during the project meetings (<http://savannah.cern.ch/task/?group=cool>). In the short term, the main priorities for functional extensions are the implementation of a deployment model involving read-only and write-only accounts with no DDL privileges, the completion of user examples and documentation, the provision of a few missing methods for basic tag manipulation, the completion of support for the SQLite backend and the possibility to store long strings as CLOBs in the database. Towards the end of 2005, the emphasis will move to the development of tools and APIs for data distribution at COOL deployments outside CERN. On the same timescale, COOL will as the other relational components migrate to the use of the new RAL package soon after its release to profit from service lookup, grid authentication, connection retry and client monitoring, and from many other functional and performance enhancements recommended by the COOL team during the RAL review.

In parallel to ongoing development of COOL the first production release is currently undergoing a series of performance and scalability tests stressing both the software and the back-end service. The tests just started with the CERN Tier 0 applying reference workloads and functional requirements as defined by the experiments for different use cases, such as online condition data population, offline condition reading. A comprehensive multi-client COOL test suite will be developed for this purpose which will be the basis for regression tests on new software releases and to validate service changes (eg new database server hardware /configuration or new database software releases). The first results from the tests already pointed out the need for some high-priority performance enhancements in COOL and in RAL, such as the simultaneous bulk

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 36	

insertion/retrieval of IOVs into/from different channels in the same COOL folder or an improved handling of Oracle cursors in RAL.

### Milestones:

ID	Delivery Date	Description
CDB01	2005/04/30	First COOL production release
<i>CDB02</i>	<i>2005/07/31</i>	<i>Define reference performance requirements for COOL software and COOL service (ATLAS online exists)</i>
<i>CDB03</i>	<i>2005/07/31</i>	<i>COOL 1.2.1 release with support for SQLight backend and Oracle tools to grant reader/writer/tagger privileges</i>
<i>CDB04</i>	<i>2005/07/31</i>	<i>COOL tier 0 deployment starts for ATLAS</i>
<i>CDB05</i>	<i>2005/08/31</i>	<i>User examples and basic user guide, including documentation of current and foreseen tag functionality</i>
<i>CDB06</i>	<i>2005/08/31</i>	<i>COOL 1.3.0 release with support for large character objects (CLOBS)</i>
<i>CDB07</i>	<i>2005/09/30</i>	<i>COOL software review</i>
CDB08	2005/10/15	COOL 1.4.0 release with bulk insertion and retrieval in different “channels” within the same “folder”
CDB09	2005/10/15	COOL passes all tier 0 validation tests
<i>CDB10</i>	<i>2005/10/15</i>	<i>COOL integration into ATLAS and LHCb offline framework finished</i>
<i>CDB11</i>	<i>2005/11/30</i>	<i>COOL 1.5.0. release based on new RAL release</i>
CDB12	2005/12/31	First prototypes of API and command line tools for data extraction and cross-population between COOL databases

Table 14 Table 15 CondDB level-2 and level-3 (italic) milestones

## 7.4 Technical process

POOL and COOL follow the LCG Application Area software process and POOL has very actively contributed to its definition. The POOL project uses a release procedure, which is based on individual component tags, which get integrated into project internal releases. These releases are picked up by the experiments for early validation of new functionality and to confirm bug fixes. Internal releases are produced relatively frequent – typically every 2 weeks. Production releases that are fully validated and supported are produced based on the experiments plans and the outcome of their validation every 2-3 month. POOL has to maintain compatibility for older releases while providing validation releases with new, still incomplete functionality. This requires the project to regularly branch-off the production release line from the development head (eg during root4 and reflex validation). The POOL developers provide documentation as part of the release

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 37

procedure. Reference and User documentation are produced in html and pdf formats based on DocBook fragments that are tagged in CVS together with the software releases. POOL and COOL are coordinated via open developer meetings, which take place every two weeks for POOL and each week for COOL. Project plans and requests for experiment prioritisation are reported to the Application Area Architect Forum. Project status and plans are regularly exposed to the larger community during experiment weeks and during persistency workshops organised at least once per year.

## 7.5 Resources and Milestones

### 7.5.1 Staffing

The following table lists active persistency framework developers – people who have recently committed code to the persistency framework repository. In case a commitment to continue their participation in 2005 has been obtained this is listed as well.

The total committed manpower for the year 2005 sums up to 7 FTE of which 3.4 FTE are contributions from LHC experiments.

Name	FTE	Work package
CERN IT / LCG		
Radovan Chytrcek	0.7	RAL, MySQL plug-in, client monitoring
Dirk Duellmann	0.5	Persistency framework leader
Giacomo Govi	0.7	RAL, object mapping
Maria Girone	0.2	Catalogs
David Front		Application & DB service validation (currently COOL performance testing)
Ioannis Papadopoulos	0.8	Persistency service, RAL, object relational storage manager, primary release coordination
Andrea Valassi	0.8	COOL subproject leader, COOL kernel
ATLAS		
Kristo Karr	0.5	Root and databases collections
Marcin Nowak	0.5	Storage Manager, schema evolution, schema loading
Sven Schmidt	0.8	COOL kernel
Antoine Perus	0.5	COOL
Uli Moosbrugger	0.4	COOL
CMS		

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 38

Zhen Xie	0.3	Catalogs, Object Relational Storage
William Tanenbaum	0.1	Root storage manager
LHCb		
Markus Frank	0.2	Storage manager, ROOT backend
Marco Clemencic	0.1	COOL, RAL

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 39	

## 8. Simulation Project

### 8.1 Purpose, Scope and Objectives

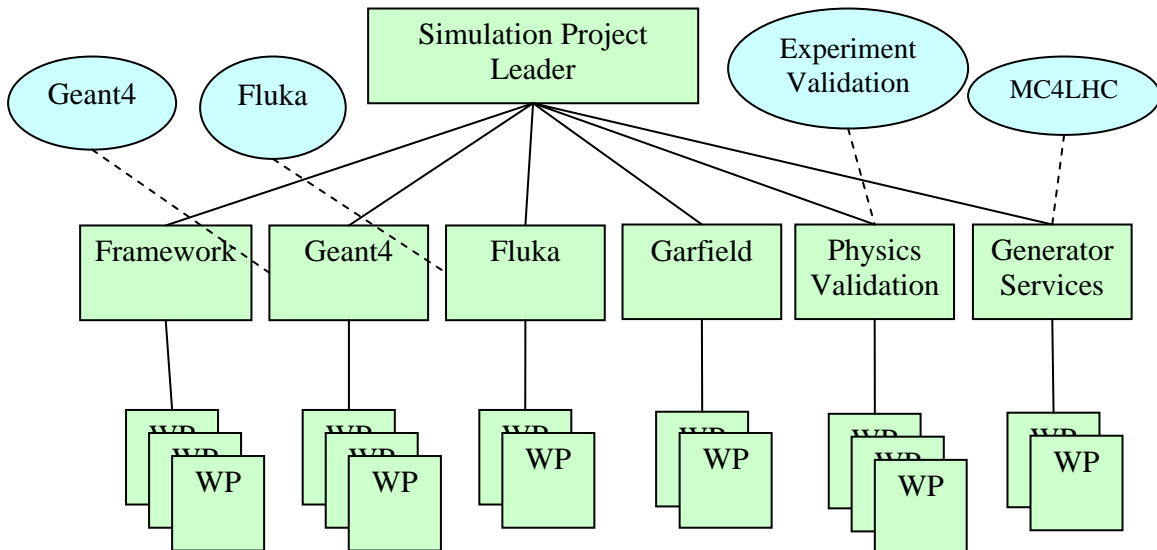
The Simulation Project covers a range of activities in the simulation as part of the LCG Applications Area, encompassing common work among the LHC experiments. Among its activities it is included the development of a simulation framework and infrastructure for physics validation studies, the CERN and LHC participation in the Monte Carlo generator services and in the Geant4, Fluka and Garfield simulation engines. The Project coordinates a major effort among all the experiments and the LCG to validate the physics of the simulation using and providing a simulation infrastructure and event generator services.

### 8.2 Project Organization

The LCG Simulation Project actively participates both in terms of management and development activities to external projects like Geant4 and Garfield and sees external participation of Fluka and Monte Carlo generators authors, as well as the participation of experiments' experts in the validation activity.

The structure in subprojects is shown in Figure 3.

The overall project is led by Gabriele Cosmo (CERN). The subprojects are led by John Apostolakis (Geant4), Paolo Bartalini (Generator Services), Alfredo Ferrari (Fluka), Witold Pokorski (Simulation Framework), Alberto Ribon (Physics Validation) and Rob Veenhof (Garfield).



**Figure 3 - The overall structure of the Simulation Project. External participations to each subproject (blue circles) are also shown.**

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 40	

### 8.3 Simulation Framework

The general task of the Simulation Framework subproject is to provide flexible infrastructure and tools for the development, validation and usage of Monte Carlo simulation applications. The purpose of this is to facilitate interaction between experiments and simulation toolkits developers as well as to eliminate duplication of work and divergence.

#### 8.3.1 WP1 - Geometry Description Markup Language

The Geometry Description Markup Language (GDML), initially developed in the context of an XML based geometry description for Geant4, has been adopted by the Simulation Project as the geometry exchange format. Its purpose is to allow interchanging detector geometries between different applications (simulation and/or analysis).

The work package consists of the GDMLSchema part, which is a fully self-consistent definition of the GDML syntax and the GDML I/O part which provides means for writing out and reading in GDML files. The GDML schema is already complete enough to support realistic geometries like the LHCb one. The implementation of some missing elements like support for division and reflections (used in some more complex geometries) is on the list of the milestones.

There has been GDML readers and writers for Geant4 supporting the present schema implemented in C++. In addition, GDML readers for Geant4 and Root have been implemented in Python.

- GDMLSchema extension to support divisions and reflections: first half of 2005
- Python implementation of GDML writers for Root: first half of 2005
- Introduction of modularization of GDML files allowing for selective importation of parts of the geometry: second half of 2005
- User support and new features implementation, such as refinement of GDMLSchema to support user extensions of elements: June 2006
- CPPGDML code revision and re-engineering to allow easier extensibility: December 2006

#### 8.3.2 WP2 - Geant4 geometry object persistency

In addition to the work on geometry interchange format, there is also some effort devoted to address direct object persistency in Geant4. It is planned to perform a feasibility study of the usage of POOL/ROOT for that purpose. Such a mechanism would be useful for running detector simulation of complex detectors, as well as for storing Geant4 geometries which are constructed interactively.

- Feasibility study of POOL/ROOT based Geant4 geometry persistency: first half of 2005

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 41

### 8.3.3 WP3 - Python interfaces to Geant4

Python interfaces to C++ applications have already proven their usefulness in adding flexibility, configurability as well as facilitating 'gluing' of different elements together. This technology has also clear benefits in the context of the detector simulation. The effort undertaken was meant to demonstrate the usage of LCG tools such as Reflex and its Python binding for running Geant4 simulation application from the Python prompt. Several examples have been implemented and documented on the Simulation Framework web page. Using the existing ROOT Python binding (PyROOT), there has also been an example implemented demonstrating Geant4 simulation interfaced to ROOT visualization, all in Python and using GDML as the geometry source.

- Working example of Python-driven Geant4 application (using Reflex): first part of 2005

### 8.3.4 WP4 – Simulation Framework for physics validation

The aim of this work package is to provide example and whenever possible, reusable code for the Flugg application in the context of Fluka-Geant4 test beam validations. Flugg is an interface which allows running Fluka with Geant4 geometry description. The activity within the Simulation Framework subproject concentrates on the application of Flugg in the context of the physics validation with a particular emphasis on the implementation of the generic interface to 'hits'.

- Example of Flugg extension applied to the test-beam validation: second part of 2005
- Completed evaluation of VMC for use in physics validation: first half of 2006

### 8.3.5 WP5 - Monte Carlo truth handling

Monte Carlo truth handling is a difficult task, especially for large multiplicity events of LHC. There is a large number of particles produced, for instance, in the showers and the selection criteria for filtering out unwanted particles are often complicated. All of the LHC experiments have come up with their own solutions, but further improvement of the performance and flexibility is possible. Some effort will be devoted within the Simulation Framework subproject to perform a feasibility study of a common mechanism for MCTruth handling.

- Proposal for universal handling of Monte Carlo truth: second half of 2005
- Tentative first release of a common framework for MCTruth handling: second half of 2006

### 8.3.6 Sub-project Milestones

ID	Delivery date	Description
SF505	2005/03/31	New release of GDML 2.1.0 with improvements in configuration scripts and kernel modules
SF506	2005/03/31	First prototype of a Python-driven Geant4 simulation using

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 42	

		LCGDict and PyLCGDict
SF510	2005/06/15	<i>New release of GDML including 'autoconf' based build system front-end</i>
SF511	2005/06/15	<i>Working example of POOL/ROOT-based geometry persistency mechanism for Geant4</i>
SF515	2005/06/30	GDML-Schema extension to support divisions and reflections
SF516	2005/06/30	Feasibility study of POOL/ROOT based Geant4 geometry persistency
SF517	2005/06/30	Python implementation of GDML writers for Geant4 and ROOT
SF520	2005/07/15	<i>First version of ROOT-&gt;GDML geometry writer in Python</i>
SF530	2005/09/30	Flugg application to one of the LHC calorimeter test-beam simulation
SF537	2005/12/15	Introduction of modularization of GDML files
SF538	2005/12/15	Proposal for universal handling of Monte Carlo truth
SF539	2005/12/15	Evaluation of VirtualMC for usage in Geant4-Fluka test-beam validation studies
SF608	2006/06/30	Refinement to GDMLSchema to support user extensions of elements
SF609	2006/06/30	Completed evaluation of VMC for use in physics validation
SF613	2006/10/31	First release of a common framework for MCTruth handling
SF615	2006/12/15	CPPGDML code revision and re-engineering to allow easier extensibility

**Table 16 Simulation Framework level-2 and level-3 (italic) milestones**

## 8.4 Physics Validation

The goal of this subproject is to compare the main detector simulation engines for LHC, Geant4 and Fluka, with experimental data, in order to understand if they are suitable for LHC experiment applications. The basic strategy for validating these simulation programs is that the dominant systematic effects for any major physics analyses should not be dominated by the uncertainties coming from simulation. This approach relies on the feedback provided by the physics groups of the LHC experiments to the developers of these simulation codes.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 43	

### 8.4.1 WP1 – Impact of simulation on LHC physics

In this work package, we aim to collect and revisit requirements for the simulation packages from the LHC experiments, by evaluating the impact of simulation uncertainties on physics observables.

- A document summarizing the major requirements has been already released, but is expected to be updated in 2006, based on the analyses of the latest (2004) test-beams. Furthermore, the current effort in the CMS collaboration for preparing the Physics TDR could also provide important new feedbacks to the simulation developers.

### 8.4.2 WP2 - Electromagnetic physics

Electromagnetics is always the first physics component of any simulation code to be validated, because it plays an important role also in hadronic showers. The goal of this work package is exactly to test the electromagnetic physics of the simulation packages, via both test-beam data (using electrons and muons as beam particles) and simple benchmarks (i.e. thin target setups).

The electromagnetic physics for Geant4 has been already validated at percent level, and there is some activity in ATLAS to push its precision below that level, allowing the use of simulation for calibrating the calorimeter.

Important new validation results of electromagnetic physics (and of hadronic physics as well, see WP3a and WP3b) will be produced by the analyses of the latest (2004) test-beams. Preliminary results are expected in the second half of 2005, whereas more complete analyses should be available in 2006.

### 8.4.3 WP3a – Hadronic physics: calorimetry

Hadronic physics is a very broad and complex field, due to the inability of the underlying theory, quantum chromo-dynamics (QCD), to make calculable, perturbative predictions for hadron-nucleus reactions in the kinematics region relevant for describing the passage of hadrons through a detector. The simulation therefore relies on different models, which need to be tested in various combinations of beam particle type, beam kinetic energy, and target material. Given the variety of aspects of hadronic physics validation, we have subdivided WP3 in three work sub-packages. In the first one described in this section, we check the hadronic physics simulation with calorimeter test-beam data. In this case, the physics observables are macroscopic properties of hadronic showers, which are the results of many interactions and effects. Simple benchmarks are also utilized to study single particle interactions, providing complementary information at microscopic level.

The first round of hadronic physics validation has been concluded with good results for the pion energy resolution and  $e/\pi$  ratio. For the longitudinal and transverse shower shapes, some discrepancies between Geant4 QGSP physics list and the data are observed. Most of the effort of LCG contribution to Physics Validation is devoted to address this issue.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 44

With version 7.0 of Geant4, some changes have been introduced to improve the shower shapes. The experiments are repeating their analyses with the old test-beam data to check if the problem is now fixed. Results should be ready by middle 2005.

The major analyses that have been produced so far with calorimeter test-beams have been carried out with the Geant4 physics lists LHEP and QGSP. It is quite interesting to compare at least one of them with Fluka. This work is also technically challenging for the Simulation Framework subproject (see below WP4). Results are expected for the second half of 2005.

After a discussion with experiments and simulation experts, the third simple benchmark was chosen: rapidity and transverse momentum distributions of inclusive charged pion produced in 100 GeV hadron (pions, kaons, protons)-nucleus (Mg, Ag, Au) interactions. Results are expected for the second half of 2005. Some other possible simple benchmarks have been identified for the future.

Important new validation results of hadronic physics in calorimeter setups (as well as in trackers, see WP3b, and in general for electromagnetic physics, see WP2) will be produced by analysing the latest (2004) test-beam data. Preliminary results are expected in the second half of 2005, whereas more complete analyses should be available in 2006.

#### **8.4.4 WP3b – Hadronic physics: inner detectors**

The second work sub-package for hadronic physics validation is focused on tests in tracking detectors. Similarly to simple benchmarks, tracking setups offer the unique possibility to test hadronic models at the level of single interactions, and, at the same time, allow to estimate the reliability of the simulation for the occupancy of the tracker and hence also for the properties of the reconstruction algorithms.

Some of the activities of this work package have been also technically relevant for the Simulation Framework subproject (see WP4).

New validation results of hadronic physics in inner detectors (as well as for calorimeters, see WP3a, and in general for electromagnetic physics, see WP2) will be produced by the analyses of the latest (2004) test-beams. Preliminary results are expected in the second half of 2005, whereas more complete analyses should be available in 2006.

#### **8.4.5 WP3c – Hadronic physics: background radiation**

The LHC caverns will be a very hostile environment, with high radiation level. Shielding is therefore mandatory to protect some components of the detectors. The design of the shielding is based on the prediction of the background radiation of some simulation codes. In spite of the use of conservative “safety factors” to scale up the predicted radiation level, and on a continuous effort to benchmark Fluka with experimental data, it is anyhow interesting to compare its results with those of other independent simulation packages, as Geant4. Differently from its predecessor, Geant3, Geant4 offers a precise treatment of low energy neutrons, which is essential for this kind of applications.

The purpose of this work package is therefore to compare the simulation results for the background radiation in the LHC experimental caverns, between Fluka and Geant4.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 45

Some preliminary results have been already presented at Physics Validation meetings by LHCb and CMS collaborations. In the former case, the tools needed to get fluencies and doses are in place, and the next step is to carefully use Geant4 with a set of “cuts” as close as possible to those used for Fluka, in order to have a consistent comparison between the two simulation engines. This is expected for the middle of 2005.

#### **8.4.6 WP4 – Validation of the simulation environment**

This last work package is somehow less physics oriented, but still very important in practice in order to have effective simulation tools available for LHC. There are two main working directions: (a) validate the shower parameterization packages (which are useful in many situations when a less accurate but significantly faster simulation is needed), and (b) to validate the adequacy and usability of simulation infrastructures (e.g. simulation frameworks, geometry description exchange, interactivity, etc.). This second part is done in close collaboration with the Simulation Framework subproject.

As far as the validation of shower parameterization packages is concerned, the plan is to start at the beginning of 2006 by reviewing the current status of these packages and the tests that have been performed so far. Then, in close collaboration with all the LHC experiments, it will be decided how to proceed. Results are expected in the second half of 2006.

For the work on Simulation Framework, an important first step has already been achieved in using Flugg to extend to Fluka the comparison with the ATLAS pixel 2001 test-beam data, starting from a Geant4 description of the setup. The next step is to achieve something similar for the much more complex case of a calorimeter test-beam setup. As described in WP3a, this work is undergoing and results are expected for the second half of 2005.

#### **8.4.7 Sub-project Milestones**

<b>ID</b>	<b>Delivery date</b>	<b>Description</b>
VD502	2005/02/28	Review/prioritization of simple benchmarks for physics validation
VD508	2005/05/15	First results of background radiation studies with Geant4
VD512	2005/06/30	New validation results on longitudinal shower shapes with Geant4 7.0 completed
VD522	2005/09/30	3 <sup>rd</sup> simple benchmark for physics validation completed
VD524	2005/09/30	Validation of Fluka against one LHC calorimeter test-beam
VD532	2005/10/31	First results of ATLAS combined and 2004 test-beams data comparisons
VD612	2006/10/31	Validation of shower parameterization packages completed

**Table 17 Physics Validation level-2 and level-3 (italic) milestones**

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 46

## 8.5 Geant4

This sub-project contributes to support/maintenance and development of Geant4 in a number of areas. The continuous response to problem reports, inquiries, requests to port to and test on new compilers, etc in these areas from users applying Geant4 in production for LHC and related areas require a significant and growing fraction of the current effort. In addition development, verification and validation are continuing to address existing and new requirements from the LHC experiments.

The work of the sub-project is sub-divided into work packages according to the work area in which support, maintenance and development are envisaged. In addition there are dedicated work-packages for the software management aspects, for the system integration testing and the coordination/release aspects.

### 8.5.1 WPI - Geometry, Field and Transportation

Undertakes leading role in the support, maintenance and further development of the corresponding Geant4 module for creating, and navigating in model detector geometries, for propagating charged tracks in the model geometries and for communicating the results to the tracking module. Additional capabilities include the provision of event-biasing, using Russian roulette and splitting, and driven by volume importance values or space-energy weight-windows. The main tasks are:

- Geometry developments including improved parameterizations and a generalized twisted trapezoid shape. Scheduled by end of March 2005.
- A revision of the geometry optimization mechanism to handle re-optimization of local sub-trees for usage in dynamic geometry setups. First design iteration for allowing navigation on parallel geometries. Evaluation and tuning/benchmark of performance for transportation in field. Scheduled by end of June 2005.
- First implementation of a new ellipsoid shape. Prototype implementation for allowing tuning of the precision/tolerance for points on the surface of solids. Scheduled by end of September 2005.
- An implementation review of the second order equations algorithms for tube and cone solids, in the attempt to increase precision in the calculation of intersections. Extensions to the optimisation mechanism for special handling of navigation in regular geometry structures. Scheduled by end 2005.
- Implementation of new tools to help in the identification of overlaps in the user's geometry at the time of construction (i.e. placement of volumes in the geometry setup). Scheduled by end of March 2006.
- First implementation of parallel navigation to enable better radiation monitoring of charged particles at arbitrary locations. Scheduled by end of May 2006.
- Implementation for tuneable tolerances applied for the treatment of points on the solids' surface, based on the implemented prototype. Scheduled by end 2006.
- First implementation of Boundary Represented Solids (BREPS) for experiment support structures. Scheduled by March 2007.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 47

### 8.5.2 WP2 - Software Management

This work package undertakes the development and maintenance of the ‘global’ category in Geant4; it is responsible for release management, creation of release distributions including binary distributions, and provides common services like a web server, CVS service, etc. to the Geant4 collaboration. Most of these tasks are ongoing support activities for which long term milestones are difficult to plan. The main tasks are:

- Move of CVS repository to CVS service provided by CERN/IT (Jan 2005).
- Move problem reporting system to Savannah, or assist the transfer of Bugzilla to another Geant4 site (Dec 2005).
- Investigate streamlining of distributions to simplify building (Mar 2005).
- Code migrations to CLHEP 2.x, gcc-3.4 and 64 bit architectures, including AMD64 (Sep 2005), sstream (Dec 2005). Ongoing work for new compiler versions and OS versions as required.
- Support for Geant4 tools, services and servers: move to new servers for Web and MySQL database (Sep 2005), review tools documentation and improve/add to existing documentation (Sep 2005), review tools with emphasis to improving security (Sep 2005).

### 8.5.3 WP3 - EM Physics and error propagation

The EM physics package in the Geant4 toolkit covers sub-packages for simulation of electromagnetic interactions of particles with matter. These ‘standard’ EM processes cover electrons and gammas ( $e^-/e^+/\gamma$  above 1 KeV), muons, x-rays (Cerenkov, transition radiation, etc.), high-energy, optical. In addition utility classes, a physics list sub-package, examples and tests are provided, maintained and refined. Key parts are the processes and models for energy loss of charged particles, for multiple scattering, and for gamma and optical photons interactions. The work is carried out by CERN group in collaboration with other Standard EM developers. The main tasks are:

- Improvements and additions to the ‘standard’ EM processes, completing the set of corrections to Bethe-Bloch model for ionization processes, and extending modeling of transition radiation (Dec 2005).
- Provide simulation of rare high energy EM processes for LHC, including radioactive corrections to major EM processes, positron annihilation to hadrons, and pair production by protons & pions for use in simulation of hadron and EM showers at LHC (Dec 2006).
- Extensions and additions to the ‘standard’ EM processes to complete the set of EM processes relevant to LHC physics. These include more precise modeling of synchrotron radiation, improvements in the LPM effect and photo-effect in light materials (Jun 2007).
- Evaluate and improve the stability and precision of shower response in LHC calorimeter when changing production thresholds. This requires evaluation of step dependence of observables due to the current methods of modeling multiple scattering and of sampling fluctuations (Jun 2006).

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 48

- Enable simulation of EM interaction of potential exotic particles at LHC (June 2005) and provide a set of specific models for exotics (Jun 2007).
- Enable configuration and steering of EM ‘package’ in applications, by providing physics list components, and new capabilities to set precise model only for a particular sub-detector or region (Dec 2005).
- Enable interfacing of existing EM models for error propagation in reconstruction - to provide estimates of energy loss and multiple scattering effects (Mar 2005) and provide a set of components for forward/backward propagation of particles in reconstruction (Dec 2006).

A continuing activity will be to provide support, maintenance, validation, acceptance testing and CPU-usage optimization of the ‘EM-standard’ package.

#### **8.5.4 WP4 - Hadronic Physics**

The hadronic physics work package contributes to the work of the hadronic physics working group in Geant4 and offers advice and support to LHC experiments on topics related to interactions of hadrons. Physics models currently supported by the hadronic physics work package include the QGS model, the Binary Cascade, the Chiral Invariant Phase Space (CHIPS) model, a model for elastic interactions of hadrons with matter, and physics lists for the LHC experiments.

The main tasks foreseen include:

- Enable background and radiation studies through the optimization of physics list for these studies (Jun 2005).
- Integrate the string fragmentation work in CHIPS with the parton string models to enable its trial use for physics performance in realistic scenarios (Jun 2005).
- Provide support for issues arising from LHC detector test-beam analyses. This includes an elaboration on the ATLAS calibration problem (Sep 2005).
- Participate in the effort to include simulation of exotic physics for ATLAS and CMS by providing assistance for the inclusion of processes for exotic particles like split super-symmetry, gravitino, the lightest SUSY particle, etc... (Sep 2005).
- Finalise paper on neutron transport and on deep inelastic scattering using CHIPS (Jul 2005).
- Investigate and resolve issues in hadronization part of QGS model concerning:
  - o Transverse momenta of secondaries produced (Dec 2005);
  - o Masses of resonances created (Mar 2006).
- Extend Binary Cascade with propagate interface to strings models (Sep 2005) and re-design parts of the Binary Cascade with the goal to encapsulate field-transitions (Sep 2006).
- Provide physics list based on CHIPS model first using QGSC for high energies (Sep 2005) and later replacing QGSC by a CHIPS quasmon-string model (2006).

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 49	

- Improve regression testing of physics performance for CHIPS model (Dec 2005).
- Study the need and provide refined physics lists with newer model options and improvements for low rate processes (Sep 2006).
- Include additional verification tests and simple benchmarks for hadronic interactions in the intermediate energy range (10-50 GeV) (Jun 2006).
- Benchmark Geant4 for the use in shielding and radiation environment applications (Sep 2006).
- Improvement of pre-compound and evaporation models (Mar 2007, pending additional manpower).

### 8.5.5 WP5 - System Testing

The system testing working group provides a service for integration testing for all the development and maintenance undertaken on the Geant4 toolkit (including source code, data and physics list). For this purpose it maintains a facility which includes tools and testing infrastructure. The tools enable developers to log their changes as directory tree tags, classify them and potentially submit them for testing, monitor testing phase and track their resulting acceptance or rejection.

The regular suite of system tests includes tests of integration and simple comparisons with restricted running times. This suite is exercised regularly on a list of supported platforms with several CPU/compiler/OS combinations, and must be successful before tags are accepted. An extended version of these tests is also run on a regular, monthly, basis to identify infrequent problems.

In addition to the ongoing service, a number of development objectives regarding the tests and the testing tools are planned. Their milestones are:

- Prototyping of 'standard notation' for results (output) (Jun 2005) and first deployment (Dec 2005).
- Clarification of coverage map for tests/examples (Jun 2006).
- Automation for keeping results and track performance of tests (Jun 2005).
- Full deployment of 'standard notation' for test results (Dec 2006).

### 8.5.6 WP6 - Acceptance Suite

The goal of this work package is to provide a set of tools for the automatic comparison of key physics observables between different versions of Geant4. The validation tests are chosen to be simplified but representative of typical LHC use cases. Running these will provide a level of assurance that the attained physics performance for typical test cases is maintained.

It undertakes a complementary role to the verification testing made by physics working groups on their separate development, and that of the system testing for integration.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 50

The main application is the systematic regression testing for Geant4 public releases, to identify potential statistically significant discrepancies in a number of physics distributions, between the new release and the established results of a chosen release. Given the large number of potential observables one could consider, it is essential to restrict as much as possible the number of those distributions that need to be looked by a person. Therefore, a variety of powerful statistical tests are necessary.

The main deliverables are:

- Improvements in the deployment on the Grid of the test suite, in the monitoring of the p-values and the CPU time (Mar 2005).
- Tools to summarize results with key statistics and plots (Jun 2005).
- Revised tests with additional observables, and extensions of the statistical tests (Sep 2005).
- Prototype for storing/retrieving previous results (n-tuples and plots) (Dec 2005).
- Extension of tests to electron projectiles, and comparison with existing electromagnetic physics validation tests (Jun 2006).
- Addition of thin-target tests for hadronic physics (Dec 2006).

### 8.5.7 WP7 - Coordination and Releases

The objectives of this work package are to coordinate work in SFT on Geant4, to collect and follow requirements from the LHC experiments, and to undertake the release management of Geant4 releases, patches and development tags.

It also assists in integration LHC requirement in the work plans for the Geant4 modules in which SFT is not involved.

### 8.5.8 Sub-project milestones

ID	Delivery date	Description
G4503	2005/03/25	Development release of Geant4 including fixes and improvements in geometry and physics. Issues discovered in production; geometry developments including improved parameterizations and generalized twisted trapezoid and a prototype of energy loss process with capability for backward extrapolation
G4518	2005/06/30	Release 7.1: refinements to ionization processes, additional string fragmentation and verification of proton reactions at high energies. Prototype of specialized model for multiple scattering for electrons; first implementation of gamma processes in model interface and of materials builder using NIST data; integration of CHIPS string fragmentation with the parton string models, enabling trial use for physics studies; verification of proton reactions at high energies
G4519	2005/07/15	Tutorial on Geant4 and move of CVS repository to IT service. This includes move to IT service for the Geant4

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 51	

		source code repository; finalised paper on neutron transport and on deep inelastic scattering using CHIPS; revised tutorials and sections of Geant4 User Documentation
G4523	2005/09/30	Development release: cascade interface to strings, stability study for EM observables and review of LPM effect. Including studies on stability of EM quantities from sampling calorimeters against changes in cuts, max-step size; review of LPM effect in EM processes; propagator interface in Binary Cascade to string models; prototype port to CLHEP 2.x.
G4531	2005/10/31	Improved regression suite for release validation and testing infrastructure. With improvements in Bonsai enabling open viewing and authorization for changes, common automation for system tests and "acceptance suite" regression tests, study of power of regression tests for shower shape
G4542	2005/12/20	New public scheduled release. It includes fixes, improvements and new developments, including positron annihilation and geometry voxelisation improvements.
G4605	2006/03/31	Development release including new tool for overlap detection at geometry construction and extensions to QGS.
G4606	2006/05/31	Development release. Including new features for parallel navigation enabling scoring charged particles at arbitrary locations, improvements to stability of showering for changes in cuts, and additional verification tests for hadrons between 10 and 50 GeV, as part of potential June 2006 public release of Geant4
G4611	2006/09/30	Geant4 development release. Including developments including redesign of Binary Cascade's field transitions, additional benchmarking for radiation and shielding use cases and refinements to physics lists for low-rate processes.
G4614	2006/12/01	Geant4 development release. Including surface tolerances tuned to model geometry size
G4701	2007/06/01	Geant4 development release. Including refined models for EM interactions of exotic particles, first implementation of tessellated BREP solids

Table 18 Geant4 level-2 milestones

## 8.6 Fluka

The Fluka team participates as external members to the Simulation Project; the collaboration is beneficial for the Simulation Framework and Physics Validation subprojects, where the Fluka simulation engine is interfaced and utilised.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 52	

The plan of work is reported in this document for completeness, although not included in the Simulation Project planning.

### 8.6.1 *Summary of milestones*

Below it is listed the set of major achievements and activities planned in Fluka for the next months:

- Completion of PEMF elimination and thorough testing of the new solution.
- Completion of the radioactive decay modules.
- Completion of the "name based" input transition.
- Final elimination of the intermediate energy hadronic model (superseded by an extension of PEANUT).
- Final setup of the new low energy neutron cross section library (production use postponed after the beta-release).
- Cleanup and proper documentation/acknowledgments of the source code.
- beta-release (source included, limited to CERN and INFN).
- Release of the first report (a second one is foreseen by the end of the year/beginning 2006) on FLUKA.
- Thorough testing and cross-checks of the new beta-version.
- Feedback to user questions/reports.
- Further development on advanced heavy ion physics features as required by LHC-ION and NASA.
- Production use of the new low energy neutron cross section library.
- Production use of the "extended" PEANUT interaction model (it will include and substitute the current high energy interaction model).

## 8.7 Garfield

Garfield is a computer program for the detailed simulation of two- and three-dimensional chambers made of wires and planes, such as drift chambers, TPCs and multi-wire counters. It accepts two and three dimensional field maps computed by finite element programs such as Maxwell, Tosca and FEMLAB as a basis for its calculations. Work is on-going to upgrade interfaces to all these finite element programs.

An interface to the Magboltz program is provided for the computation of electron transport properties in nearly arbitrary gas mixtures. Garfield also has an interface with the Heed program to simulate ionisation of gas molecules by particles traversing the chamber. New releases of both Heed and Magboltz are in the process of being interfaced. The integration of the new release of Heed will also mark a major change in the programming aspects of Garfield since Heed is now written in C++.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 53

Transport of particles, including diffusion, avalanches and current induction is treated in three dimensions irrespective of the technique used to compute the fields. Currently Monte-Carlo simulations of drift with diffusion assume Gaussian spreads. This is not applicable in detectors such as GEMs where, indeed, the calculated diffusion spread depends on the step length. Correcting this is in progress.

Negative ion TPCs are being considered as detectors in the search for dark matter. To simulate these devices not only needs attachment processes, which are already available, but also dissociation processes. These are in the process of being written.

### 8.7.1 WP1 – Electric field and Diffusion

Future developments include:

- Interfaces with Maxwell 2D (Ansoft) and FEMLAB, work on the interface for Tosca, by February 2005.
- Diffusion in strongly converging and diverging fields in GEM and Micromegas detectors, by April 2005.

### 8.7.2 WP2 – Gas and Transport

Future developments include:

- Interface with a new version of Magboltz, by end of 2005.
- Interface with a new version of Heed, by end of 2005.

### 8.7.3 Sub-project milestones

ID	Delivery date	Description
GF501	2005/01/31	Interfaces with Maxwell 2D (Ansoft) and FEMLAB, work on the interface for Tosca
GF507	2005/03/31	Diffusion in strongly converging and diverging fields (GEM and Micromegas)
<i>GF509</i>	<i>2005/05/16</i>	<i>Lecture series for graduate students in Cagliari</i>
<i>GF521</i>	<i>2005/09/10</i>	<i>Invited talk at TRD 2005 conference (Ostuni, Italy) on photon detection</i>
GF540	2005/12/15	Interface with a new version of Magboltz
GF541	2005/12/15	Interface with a new version of Heed

Table 19 Garfield level-2 and level-3 (italic) milestones

## 8.8 Generator Services

The Generator services project collaborates with Monte Carlo (MC) generators authors and with LHC experiments in order to prepare validated LCG compliant code for both the theoretical and experimental communities at the LHC, sharing the user support duties,

Organisation CERN – LCG Project		Title/Subject Applications Area Plans Phase 2	Number	
Owner	Approved by	Date 26/09/2005	Version 1.2	Page 54

providing assistance for the development of the new object oriented generators and guaranteeing the maintenance of the older packages on the LCG supported platforms.

Contact persons for most of MC generator packages relevant for the LHC and representatives for all the LHC experiments have been agreed.

Responsibilities are currently shared in the following way: coordination (0.30 FTE, University of Florida); liaison with experiments & Monte Carlo project (0.25 FTE, CERN), WP1-librarian (0.25 FTE, CERN), WP1-integration (0.75 FTE, LCG Russia), WP3-MCDB (0.75 FTE, LCG Russia), WP3-Production (0.5 FTE, University of Oviedo and University of Santander). On top of the activities developed internally, the coordination with other external projects allows to cover the entire spectrum of topics outlined in the work packages: WP4-validation (CEDAR/JETWEB), WP1-new object oriented MC generators (THEPEG, PHENOGRID/HERWIG++).

### **8.8.1 WP1 - The Generator library (GENSER)**

GENSER is the central code repository for MC generators and generator tools. It was the first CVS repository in the LCG simulation project. The sources and the binaries are installed in AFS and the tar-balls are made available by SPI.

The size of GENSER is growing quickly. This is currently felt as a major problem for the distribution of main and bug fix releases. The LHC experiments have submitted a specific request to allow for lighter GENSER distributions, giving access to the single GENSER sub-packages. A strategy to address the problem is in place with the help of the LCG librarian. A corresponding milestone has been set for this to be achieved by September 2005. Some of the generators supported in GENSER are already installed in the LCG external area. However, test suites and examples are always available in GENSER. .

### **8.8.2 WP2 - Storage, Event Interfaces and Particle Services**

The goal of this work package is to contribute to the definition of the standards for generator interfaces and formats, collaborating in the development of the corresponding application program interfaces.

In order to favour the adoption of the new object oriented MC generators in the experiment simulation frameworks, the Generator project will share some responsibilities on the development and maintenance of the Toolkit for High Energy Physics Event Generation (ThePEG). A common milestone with the PHENOGRID initiative for Q3 2005: the first test of ThePEG and EvtGenLHC integration in Herwig++.

### **8.8.3 WP3 - Public Event Files and Monte Carlo Database**

The goal of this work package is to produce “certified” public generator event files to be used for benchmarks, comparisons and combinations. The format and the structure of the files have to be accessible to the simulation frameworks of the LHC experiments. Three different activities have been started:

- 1) The development of a simple production and validation framework at generator level. A common software project between LCG and CMS has been defined which is relying on HepMC (event interface), ROOT and POOL (event storage).

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date	<b>26/09/2005</b>	<b>Version 1.2</b>	Page 55

Connection with the Monte Carlo data base (WP3) for configuration and book keeping is currently evaluated. The beta version of the framework should be available end Q2 2005.

- 2) A dedicated production centre integrated in the grid middleware to provide the LHC experiments and the other end-users with a transparent access to the public event files. This will be essential for those samples requiring a huge amount of CPU time and parallelization techniques.
- 3) The LCG Monte Carlo Data Base (MCDB) [[hep-ph/0404241](#)], that will be a public database for the configuration, book-keeping and storage of the generator level event files. The generator events often need to be prepared and documented by Monte-Carlo experts. MCDB aims at facilitating the communication between Monte-Carlo experts and end-users. Its use can be optionally extended to the official event production of the LHC experiments. The LCG MCDB relies on the following software packages supported by LCG: MySQL; CASTOR (RFIO); CGI; Perl; Apache. A beta prototype is currently in production on a dedicated web server: <http://mcdm.cern.ch>; authentication is currently based on AFS tokens. Work is ongoing to improve the documentation and to support authentication through GRID certificates. The MCDB team is available to provide assistance for the development of MCDB API to the experiment dependent software.

#### 8.8.4 WP4: Monte Carlo Validation

The activity in this work package is currently concentrating on the functional validation of the generator packages inserted in GENRES. If JetWeb will adopt GENRES, possible usage of JetWeb for physics validation will be considered.

#### 8.8.5 Sub-project milestones

ID	Delivery date	Description
GS504	2005/03/31	First C++ Monte Carlo (SHERPA) fully integrated in GENRES
GS513	2005/06/30	First test of ThePEG integration in Herwig++
GS514	2005/06/30	Generator level production framework beta version
GS525	2005/09/30	MCDB user document with definition of procedures to gain access through certificates
GS526	2005/09/30	Pythia 8: release of alpha version
GS527	2005/09/30	Procedure for light GENRES releases
GS528	2005/09/30	Definition of EvtGen development plans & policy: agreement on responsibilities for EvtGen development in GENRES
GS529	2005/09/30	First introduction of NRQCD Prompt Quarkonia Production models in Pythia 6.3

Organisation CERN – LCG Project		Title/Subject Applications Area Plans Phase 2		Number	
Owner	Approved by	Date 26/09/2005	Version 1.2	Page 56	

GS533	2005/09/30	<i>Proposal for a LCG Generator production centre integrated in the grid-middleware</i>
GS534	2005/12/15	<i>Integration of GENSER in JetWeb</i>
GS535	2005/12/15	Generator level production framework: production quality release
GS536	2005/12/15	<i>Evaluation of possible migration of HEPMC to GENSER</i>
GS601	2006/03/31	<i>Porting of most GENSER Fortran packages to gcc-4.0</i>
GS602	2006/03/31	Introduction of MCDB Grid certificates and Management of large files
GS603	2006/03/31	<i>Finalization of NRQCD Prompt Quarkonia Production models in Pythia 6.3</i>
GS604	2006/03/31	<i>MCDB Integration, experiment specific APIs</i>
GS607	2006/06/30	<i>Generator level validation framework beta version</i>
GS610	2006/09/30	<i>Pythia 8. Release of beta version</i>
GS616	2006/12/15	Fully operational LCG Generator production centre integrated in the grid-middleware
GS701	2007/06/30	Generator level validation framework production version
GS702	2007/09/30	Pythia 8. Release of production version

**Table 20 Generator Services level-2 and level-3 (italic) milestones**

## 8.9 Technical process

The various subprojects in the LCG Simulation Project make extensive usage of the infrastructure provided by SPI. Software packages in the Generator Services (GENSER) are released, distributed and installed following tools and standards available through SPI. Some of the available tools for Q/A are also used by the Geant4 project, the Geant4 CVS repository is planned to migrate in 2005 to the central LCG CVS facility, and a proposal for migrating the existing bug-report system to the Savannah portal is ongoing. Validation benchmarks and code for the test-beam setups and Simulation Framework packages are progressively made available in the LCG CVS repository, browsable through the tools provided by SPI.

## 8.10 Resources

### 8.10.1 Staffing

A table of the currently available and required (wished) manpower for the LCG Simulation Project is reported in Table 21

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>	Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 57

<b>Subproject</b>	<b>Available</b>
Simulation Framework	0.90
Geant4	7.70
Coordination	- 1.20
Geometry, Field, Biasing	- 2.00
Hadronic Physics	- 2.00
Electromagnetic Physics	- 1.25
Testing and Software Management	- 1.35
Physics Validation	1.50
Generator services	2.70
Management, documentation/release	- 0.70
GENSER development, validation, MCDB	- 1.75
Production Framework	- 0.25
Garfield	1.00
Management	0.25

**Table 21 - Currently available manpower for the Simulation Project. For Geant4 and Generator Services subprojects, the approximate current manpower in different areas is shown.**

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 58

## 9. Applications Area Summary

We summarize here the resources and milestones for all the Applications Area projects.

### 9.1 Resources

FTE		Affiliation										Grand Total
Project	Sub-Project	ALICE	ATLAS	BNL	CERN	CMS	EGEE	FNAL-Russia	3-Spain	LHCb	Other	
<b>MGR</b>	(blank)				0.9							0.9
<b>MGR Total</b>					<b>0.9</b>							<b>0.9</b>
<b>POOL</b>	Catalog				0.2	0.3						0.5
	Collections	0.5										0.5
	Cool	1.7			0.8					0.1		2.6
	Mgr				0.5							0.5
	Ral				2.2							2.2
	StorageMgr	0.5				0.1				0.2		0.8
<b>POOL Total</b>		<b>2.7</b>			<b>3.7</b>	<b>0.4</b>				<b>0.3</b>		<b>7.1</b>
<b>ROOT</b>	Base	0.2			0.5			0.1			0.1	0.9
	Dictionary		0.2		0.6			0.6			0.2	1.6
	Geom	0.8										0.8
	Graf	0.2			2						0.8	3
	Gui			0.1	1.3						0.2	1.6
	I/O							0.5		0.3		0.8
	Math				2.15						0.1	2.25
	Mgr				1							1
	Proof	0.4			2						0.2	2.6
	Seal				0.2							0.2
<b>ROOT Total</b>		<b>1.6</b>	<b>0.2</b>	<b>0.1</b>	<b>9.75</b>			<b>1.2</b>		<b>0.3</b>	<b>1.6</b>	<b>14.75</b>
<b>SIMU</b>	Framework				0.9							0.9
	Garfield				1							1
	Geant4				7.7							7.7
	Genser				0.45	0.25		1.75	0.25			2.7
	Mgr				0.25							0.25
	Validation				1.5							1.5
<b>SIMU Total</b>					<b>11.8</b>	<b>0.25</b>		<b>1.75</b>	<b>0.25</b>			<b>14.05</b>
<b>SPI</b>	(blank)				3.95			2				5.95
<b>SPI Total</b>					<b>3.95</b>			<b>2</b>				<b>5.95</b>
<b>Grand Total</b>		<b>1.6</b>	<b>2.9</b>	<b>0.1</b>	<b>30.1</b>	<b>0.65</b>	<b>2</b>	<b>1.2</b>	<b>1.75</b>	<b>0.25</b>	<b>0.6</b>	<b>42.75</b>

### 9.2 Milestones

#### 9.2.1 Level-1 proposed milestones

ID	Delivery date	Description
AA0	2005/03/31	Development, support and resource plan through 2008
AA1	2005/09/30	Phase 1 AA software complete and deployed
AA2	2005/09/30	ROOT workshop at CERN. Various prototypes addressing different topics of the SEAL+ROOT merge will be completed and detailed plans will be made available to the wider ROOT community.

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	Version <b>1.2</b>	Page <b>59</b>	

AA3	2006/09/30	Completed the migration to the merged SEAL and ROOT projects. End-users will not need to use anymore any of the old SEAL libraries; all the required functionality will be available in ROOT.
-----	------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 9.2.2 Level-2 proposed milestones

ID	Due	Summary
SPI01	31/10/2005	CMT and SCRAM generated automatically from XML
SPI02	30/11/2005	Independent build and release of partitions of the LCG software
SPI03	31/07/2005	Pacman caches for binaries and sources
SPI04	31/10/2005	Automated distribution system in place, generate from XML description External Software.
SPI07	30/11/2005	Import/export features available in the Savannah system. Provide mechanism to import/migrate external data into savannah.
SPI08	31/12/2005	Savannah user forum setup, user and admin documentation available.
SPI09	31/10/2005	QA reporting on savannah and test coverage available and documented.
SPI10	30/09/2005	Testing tools updated to the latest versions on the AFS area and available to LCG projects and outside.
SPI11	01/08/2005	Doxygen and LXR automated and documented for the LCG software and for the packages (ROOT, AIDA, CLHEP)
BAS1	01/08/2005	New updated plugin manager
BAS2	15/07/2005	A new edition of the ROOT Users Guide
BAS3	15/07/2005	The ROOT bug tracking system moved to Savannah
DIC1	30/06/2005	Reflex and Cintex in the ROOT v5 development release
DIC4	31/12/2005	CINT fully converted to Reflex
DIC5	31/03/2006	PyRoot exclusively based on Reflex
PRF7	30/09/2005	Demonstration of new PROOF system at ROOT 2005
MAT1	30/06/2005	First version of new MathCore library
MAT2	31/08/2005	First version of the MathMore and random number generator library
MAT3	30/09/2005	New C++ Minuit in be released in ROOT
MAT4	31/12/2005	Finalization of the Fitting and Minimization interfaces
MAT5	31/12/2005	RootFit fully integrated (in CVS by July)
MAT6	32/12/2005	Evaluation of a prototype linear algebra package independent of TObject
GRA1	30/09/2005	Demonstrate the new GL viewer at the ROOT workshop.
SEAL1	30/11/2005	Remove from new SEAL releases Reflex and Cintex packages. Bug fix releases will continue to include them.
SEAL2	30/11/2005	Remove from new SEAL releases Minuit and MathCore. Bug fix releases will continue to include them
SEAL3	31/12/2005	Remove from new SEAL releases Reflection, ReflectionBuilder and the Scripting packages (PyLCGDict and PyLCGDict2)

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 60	

POOL01	31/03/2006	Reflex production release (Reflex validation by experiments completed)
POOL02	31/03/2006	Overall performance study and validation with expected requirements
RAL03	31/08/2005	RAL validation release (implementing the RAL review outcome and separate packaging)
RAL05	31/10/2005	RAL production release
RAL06	30/11/2005	Production release of proxy-cache access plugin
CAT02	31/08/2005	Production release of LCG catalog adaptors (level 2)
CDB01	30/04/2005	First COOL production release
CDB08	15/10/2005	COOL 1.4.0 release with bulk insertion and retrieval in different "channels" within the same "folder"
CDB09	15/10/2005	COOL passes all tier 0 validation tests
CDB12	31/12/2005	First prototypes of API and command line tools for data extraction and cross-population between COOL databases
SF505	31/03/2005	New release of GDML 2.1.0 with improvements in configuration scripts and kernel modules
SF506	31/03/2005	First prototype of a Python-driven Geant4 simulation using LCGDict and PyLCGDict
SF515	30/06/2005	GDML-Schema extension to support divisions and reflections
SF516	30/06/2005	Feasibility study of POOL/ROOT based Geant4 geometry persistency
SF517	30/06/2005	Python implementation of GDML writers for Geant4 and ROOT
SF530	30/09/2005	Flugg application to one of the LHC calorimeter test-beam simulation
SF537	15/12/2005	Introduction of modularization of GDML files
SF538	15/12/2005	Proposal for universal handling of Monte Carlo truth
SF539	15/12/2005	Evaluation of VirtualMC for usage in Geant4-Fluka test-beam validation studies
SF608	30/06/2006	Refinement to GDMLSchema to support user extensions of elements
SF609	30/06/2006	Completed evaluation of VMC for use in physics validation
SF613	31/10/2006	First release of a common framework for MCTruth handling
SF615	15/12/2006	CPPGDML code revision and re-engineering to allow easier extensibility
VD502	28/02/2005	Review/prioritization of simple benchmarks for physics validation
VD508	15/05/2005	First results of background radiation studies with Geant4
VD512	30/06/2005	New validation results on longitudinal shower shapes with Geant4 7.0 completed
VD522	30/09/2005	3rd simple benchmark for physics validation completed
VD524	30/09/2005	Validation of Fluka against one LHC calorimeter test-beam
VD532	31/10/2005	First results of ATLAS combined and 2004 test-beams data comparisons
VD612	31/10/2006	Validation of shower parameterization packages completed

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 61	

G4503	25/03/2005	Development release of Geant4 including fixes and improvements in geometry and physics. Issues discovered in production; geometry developments including improved parameterizations and generalized twisted trapezoid and a prototype of energy loss process with capability for backward extrapolation
G4518	30/06/2005	Release 7.1: refinements to ionization processes, additional string fragmentation and verification of proton reactions at high energies. Prototype of specialized model for multiple scattering for electrons; first implementation of gamma processes in model interface and of materials builder using NIST data; integration of CHIPS string fragmentation with the parton string models, enabling trial use for physics studies; verification of proton reactions at high energies
G4519	15/07/2005	Tutorial on Geant4 and move of CVS repository to IT service. This includes move to IT service for the Geant4 source code repository; finalised paper on neutron transport and on deep inelastic scattering using CHIPS; revised tutorials and sections of Geant4 User Documentation
G4523	30/09/2005	Development release: cascade interface to strings, stability study for EM observables and review of LPM effect. Including studies on stability of EM quantities from sampling calorimeters against changes in cuts, max-step size; review of LPM effect in EM processes; propagator interface in Binary Cascade to string models; prototype port to CLHEP 2.x.
G4531	31/10/2005	Improved regression suite for release validation and testing infrastructure. With improvements in Bonsai enabling open viewing and authorization for changes, common automation for system tests and "acceptance suite" regression tests, study of power of regression tests for shower shape
G4542	20/12/2005	New public scheduled release. It includes fixes, improvements and new developments, including positron annihilation and geometry voxelisation improvements.
G4605	31/03/2006	Development release including new tool for overlap detection at geometry construction and extensions to QGS.
G4606	31/05/2006	Development release. Including new features for parallel navigation enabling scoring charged particles at arbitrary locations, improvements to stability of showering for changes in cuts, and additional verification tests for hadrons between 10 and 50 GeV, as part of potential June 2006 public release of Geant4
G4611	30/09/2006	Geant4 development release. Including developments including redesign of Binary Cascade's field transitions, additional benchmarking for radiation and shielding use cases and refinements to physics lists for low-rate processes.
G4614	01/12/2006	Geant4 development release. Including surface tolerances tuned to model geometry size
G4701	01/06/2007	Geant4 development release. Including refined models for EM interactions of exotic particles, first implementation of tessellated BREP solids
GF501	31/01/2005	Interfaces with Maxwell 2D (Ansoft) and FEMLAB, work on the interface for Tosca
GF507	31/03/2005	Diffusion in strongly converging and diverging fields (GEM and Micromegas)
GF540	15/12/2005	Interface with a new version of Magboltz
GF541	15/12/2005	Interface with a new version of Heed
GS535	15/12/2005	Generator level production framework: production quality release

Organisation <b>CERN – LCG Project</b>		Title/Subject <b>Applications Area Plans Phase 2</b>		Number	
Owner	Approved by	Date <b>26/09/2005</b>	<b>Version 1.2</b>	Page 62	

GS602	31/03/2006	Introduction of MCDB Grid certificates and Management of large files
GS616	15/12/2006	Fully operational LCG Generator production centre integrated in the grid-middleware
GS701	30/06/2007	Generator level validation framework production version
GS702	30/09/2007	Pythia 8. Release of production version

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 63

## 10. References

- [1] CERN-LCGAPP-2002-09 - [Architecture Blueprint RTAG report](#)
- [2] CERN-LCG-2003-029 - [Detector Simulation \(RTAG-10\)](#)
- [3] CERN-LCG-2002-024 - [Monte Carlo Generators \(RTAG-9\)](#)
- [4] CERN-LCG-2002-023 - [Detector Geometry and Material Description \(RTAG-7\)](#)
- [5] CERN-LCGAPP-2005-03 - [Applications Area internal review 2005 final report](#)
- [6] [Persistency Framework \(RTAG-1\)](#)

Organisation <b>CERN – LCG Project</b>	Title/Subject <b>Applications Area Plans Phase 2</b>	Number
Owner	Approved by	Date <b>26/09/2005</b>
		<b>Version 1.2</b>   Page 64