



DELIVERABLE 5.2.1: QUALITY ASSURANCE PLAN

2ND VERSION - DRAFT

WP5

Document Filename:	CG5.2-D5.2.1-v3.0-CYF055- QualityAssurancePlan.doc
Work package:	WP5
Partner(s):	CYFRONET
Lead Partner:	CYFRONET
Config ID:	CG5.2-D5.2.1-v3.0-CYF055-QualityAssurancePlan
Document classification:	CONFIDENTIAL

Abstract: This document specifies the CrossGrid Quality Assurance Policy together with processes for all phases of software development. The Project's structure is described along with its managerial and technical bodies; the document then presents a description of the project phases, the persons responsible for CrossGrid Quality Assurance and their roles. For each phase the relevant Quality Assurance artifacts and procedures are described. In addition, the Quality Assurance Plan discusses issues managerial reviews and other audits conducted during the Project's duration. This document extends and integrates the experience of CrossGrid and other similar projects with Quality Assurance and builds on other related documents, such as the Standard Operating Procedures document, the Software Requirement Specification documents, Design Specifications etc. - all part of the CrossGrid deliverable chain.

Delivery Slip

	Name	Partner	Date	Signature
From				
Verified by				
Approved by				

Document Log

Version	Date	Summary of changes	Author
1.0	01/04/2002	First version	Włodzimierz Funika
2.0	01/04/2003	Draft of the second version	Marios Dikaiakos, Marek Garbacz, Piotr Nowakowski
2.1	02/04/2003	Updates and extensions in most chapters and sections	Piotr Nowakowski, Robert Pająk
2.2	03/04/2003	Major updates and corrections in the first four chapters	Maciej Malawski, Piotr Nowakowski, Katarzyna Zając
2.3	23/04/2003	Minor editorial corrections and updates	Holger Marten, Piotr Nowakowski, Roland Wismüller
2.4	27/04/2003	Updates, new section on Quality Indicators	Piotr Nowakowski
2.5	03/05/2003	Major overhaul	Marian Bubak, Piotr Nowakowski, Dick Van Albada
2.6	07/05/2003	Major updates to the sections on Quality Indicators and Project Management; other minor updates, typo fixes and new graphics	Marian Bubak, Grzegorz Młynarczyk, Piotr Nowakowski, Robert Pająk
2.7	12/05/2003	More corrections and updates	Marian Bubak, Grzegorz Młynarczyk, Piotr Nowakowski
2.8	13/05/2003	Updates to section on Quality Indicators	Marian Bubak, Maciej Malawski, Grzegorz Młynarczyk, Piotr Nowakowski, Michał Turała
2.9	16/05/2003	Minor editorial updates, new managerial chart	Piotr Nowakowski
3.0	03/06/2003	Inclusion of reviewer comments	Piotr Nowakowski

CONTENTS

REFERENCES.....	5
LIST OF ABBREVIATIONS.....	6
EXECUTIVE SUMMARY.....	7
1. INTRODUCTION.....	9
2. QUALITY OBJECTIVES.....	10
3. PROJECT PRESENTATION.....	12
3.1. PROJECT OBJECTIVES AND GOALS.....	12
3.2. PROJECT STRUCTURE AND ITS RELATIONS TO QUALITY ASSURANCE.....	12
3.2.1. <i>Work Package WP1: CrossGrid Application Development</i>	13
3.2.2. <i>Work Package WP2: Application Development Support</i>	13
3.2.3. <i>Work Package WP3: New Grid Services and Tools</i>	14
3.2.4. <i>Work Package WP4: International Testbed Organization</i>	15
3.2.5. <i>Work Package WP5: Project Management</i>	15
3.3. PROJECT EVOLUTION.....	16
3.4. PROJECT MANAGEMENT STRUCTURE.....	17
3.4.1. <i>Managing of Work Packages</i>	17
3.4.2. <i>Central management and coordination</i>	18
3.4.3. <i>Steering Group</i>	19
3.4.4. <i>Collaboration Board</i>	20
3.4.5. <i>Technical Board</i>	20
3.4.6. <i>Central Dissemination</i>	20
3.4.7. <i>QA Officers</i>	21
4. DOCUMENTATION AND PUBLICATIONS.....	23
4.1. DOCUMENT STRUCTURE.....	23
4.2. DELIVERABLE REVIEW AND SUBMISSION PROCEDURE.....	24
4.3. CROSSGRID PUBLICATION POLICY.....	26
4.3.1. <i>Types of publications</i>	26
4.3.2. <i>Where to publish</i>	26
4.3.3. <i>How to publish</i>	26
4.3.4. <i>Who can publish what</i>	27
4.3.5. <i>Acknowledgements and standard phrases</i>	27
4.4. CROSSGRID DOCUMENT REPOSITORY.....	28
5. STANDARDS, PRACTICES, CONVENTIONS AND METRICS.....	29
5.1. CROSSGRID STANDARD OPERATING PROCEDURES.....	29
5.2. SOFTWARE REQUIREMENT SPECIFICATIONS.....	30
5.3. DESIGN SPECIFICATIONS.....	30
5.4. QUALITY INDICATORS.....	30
5.4.1. <i>Process Quality Indicators</i>	31
5.4.2. <i>Code development Quality Indicators</i>	31
5.4.3. <i>Testbed Quality Indicators</i>	33
6. TESTS, REVIEWS AND AUDITS IN THE CROSSGRID TIMELINE.....	34
6.1. INITIAL PHASE: SOFTWARE REQUIREMENTS REVIEW.....	34
6.2. FIRST DEVELOPMENT PHASE: DESIGN REVIEW.....	35
6.3. ALL DEVELOPMENT PHASES: IMPLEMENTATION AND SOFTWARE RELEASES.....	36
6.4. ALL DEVELOPMENT PHASES: UNIT TESTS.....	38

6.5. ALL DEVELOPMENT PHASES: TESTBED RELEASES AND VALIDATION.....	41
6.5.1. <i>Testbed release description</i>	41
6.5.2. <i>Formal steps in the preparation of a new testbed release</i>	43
6.6. ALL PHASES: PUBLICATION REVIEWS.....	44
6.7. ALL PHASES: MANAGERIAL REVIEWS	44
6.7.1. <i>Monthly Reports</i>	44
6.7.2. <i>Quarterly Reports</i>	45
6.7.3. <i>Yearly Reviews</i>	46
7. CROSSGRID CORPORATE IDENTITY	47
8. CLOSING REMARKS.....	48
8.1. APPLICATION AREA	48
8.2. QUALITY ASSURANCE PLAN EVOLUTION PROCEDURE	48
8.3. LACK OF ADHERENCE TO PROJECT QUALITY ASSURANCE.....	48

REFERENCES

- [ANNEX] CrossGrid Technical Annex; http://www.eu-crossgrid.org/CrossGridAnnex1_v31.pdf
- [ARCH] CrossGrid Deliverable D5.2.2 (CrossGrid Architecture Requirements and First Definition of Architecture); multipart document stored at <http://www.eu-crossgrid.org/M3deliverables.htm>
- [BASSIN] K. Bassin, S. Biyani, and P. Santhanam, *Metrics to evaluate vendor-developed software based on test case execution results* (IBM System Journal, Vol 41, No 1, 2002)
- [BRAUDE] Eric J. Braude, *Software Engineering: An Object-Oriented Perspective* (John Wiley & Sons, Inc., 2001)
- [CGPORTAL] The CrossGrid Web Pages; www.eu-crossgrid.org
- [DESIGN2] CrossGrid Deliverable D2.2 (Work Package 2 Design Documents); confidential
- [DESIGN3] CrossGrid Deliverable D3.2 (Work Package 3 Design Documents); confidential
- [EDG-MGMT] The DataGrid Project Management Website; <http://eu-datagrid.web.cern.ch/eu-datagrid/WP12/default.htm>
- [GRIDLAB-QA] The GridLab Project Quality Assurance Plan, available at <http://www.gridlab.org/Project/Deliverables.html>
- [GRIDPORTAL] The CrossGrid software repository and portal; <http://gridportal.fzk.de>
- [IEEE] The IEEE Software Engineering Standards catalog; http://standards.ieee.org/catalog/olis/arch_se.html
- [JONES] T. Capers Jones, *Applied Software Measurement: Assuring Productivity and Quality* (McGraw Hill, 1996)
- [PLAN] CrossGrid Deliverable D4.1 (Detailed Planning for Testbed Setup); multipart document stored at <http://www.eu-crossgrid.org/M3deliverables.htm>
- [SOP] CrossGrid Deliverable D5.2.3 (Standard Operating Procedures); <http://www.eu-crossgrid.org/Deliverables/M6pdf/CG5.2-D5.2.3-v1.0-CYF020-StandardOperatingProcedures.pdf>
- [SRS2] CrossGrid Deliverable D2.1 (Work Package 2 Software Requirements Specification); confidential
- [SRS3] CrossGrid Deliverable D3.1 (Work Package 3 Software Requirements Specification); confidential
- [STAT] CrossGrid Deliverable D4.3 (WP4 Internal Status Report); confidential
- [TATWEB] The Technical Architecture Team website; <http://kinga.cyf-kr.edu.pl/~tat/>
- [TEST] CrossGrid WP4.1 appendix 4; <http://www.eu-crossgrid.org/Deliverables/M3pdf/CG-4-D4.1-004-TEST.pdf>

LIST OF ABBREVIATIONS

AT	Architecture Team
ATL	Architecture Team Leader
CAF	Corrective Action Form
CB	Collaboration Board
CGO	CrossGrid Office
CGRB	CrossGrid Resource Broker
CGSA	CrossGrid Scheduling Agents
VS	Concurrent Versioning System
EC	European Commission
EDG	European DataGrid
EU	European Union
GPKD	Grid Portal Development Toolkit
I/O	Input/Output
IRB	Internal Review Board
IRF	Internal Review Form
LM	Local Monitor
MPI	Message Passing Interface
PC	Project Coordinator
PM	Person-Month
PO	CrossGrid Office
PS	Project Secretary
QA	Quality Assurance
QI	Quality Indicator
QAO	Quality Assurance Officer
QAP	Quality Assurance Plan
QE	Quality Engineer
RPM	Red Hat Packet Manager
SM	Service Manager
SRS	Software Requirements Specification
TAT	Technical Architecture Team
UML	Unified Modelling Language
VRVS	Virtual Reality Videoconferencing System
WP	Work Package

EXECUTIVE SUMMARY

This document constitutes the CrossGrid Quality Assurance Plan, covering all aspects related to Quality Assurance in the Project. The plan describes in detail the quality assurance methodology and guiding rules together with processes supporting their realization in all phases of the software development process, from user requirements analysis to integration testing. The current version of the Quality Assurance Plan bases on previous attempts at drafting integrated QA procedures for CrossGrid (such as the IEEE-compliant SRS templates, Design Specifications and the deliverable review procedure), and on the experiences of other affiliated European Grid projects, most notably the DataGrid Project ([EDG-MGMT]) and GridLab ([GRIDLAB-QA]).

This document follows the standard Quality Assurance Plan template IEEE 730-1989 (see [IEEE]) and is divided into the following chapters:

- Chapter 1 is an introduction to the QA process. The term is explained and the standard approach to be followed by this document is presented.
- Chapter 2 lists the CrossGrid quality objectives and characterizes their role in software development. It also provides references to the following chapters.
- Chapter 3 is a presentation of the Project along with its organizational structure. Among the elements detailed are: project Work Packages (WPs), project tasks (the basic organizational unit of the project) and project management. The chapter provides information on the following officials and bodies:
 - Project Coordinator
 - Project Secretary
 - CrossGrid Office
 - Architecture Team
 - Technical Board
 - Central Dissemination
 - Collaboration Board
 - Internal Review Board
 - Work Package Managers
 - Task Leaders
 - Quality Assurance Officers

Among the information presented are generic QA procedures to which each Work Package should adhere.

- Chapter 4 is devoted to project documentation expressed in the form of deliverables and to the publications produced by CrossGrid Partners. It describes both the physical layout of CrossGrid deliverables and the deliverable submission and acceptance procedure, involving the Internal Review Board, the CrossGrid Office and the Quality Assurance Officers. It also contains the official CrossGrid Publication Policy.
- Chapter 5 lists generic conventions, standards and metrics which the Project adheres to. Special emphasis is placed on a separate CrossGrid deliverable (D5.2.3, *Full definition of CrossGrid Standard Operating Procedures and Structure of Deliverables*). This deliverable contains extensive information regarding various technical aspects of the Project, including:

- tools used within CrossGrid (for publication as well as programming),
- coding conventions for all programming languages used within CrossGrid,
- the Central Repository and its means of usage,
- the release preparation process,
- other standardized conventions to be followed.

A separate section of this chapter is devoted to quality indicators, which will be utilized within CrossGrid.

- Chapter 6 details the CrossGrid software development process and describes the role of QA procedures at each step of the Project's timeline. This involves both the technical aspects of the project as well as the reporting that goes with it and is divided into the following sections:
 - the software requirements review (initial phase),
 - the software design review (first development phase),
 - unit testing procedures (all development phases),
 - release and prototype reviews (all development phases),
 - testbed QA (all development phases),
 - publication reviews (all phases),
 - managerial reviews (all phases).
- Chapter 7 characterizes the CrossGrid corporate identity as a set of rules, captions and graphical elements to be used in official CrossGrid publications.
- Chapter 8 contains closing remarks

This document is meant to be distributed in conjunction with the CrossGrid Standard Operating Procedures specification (see [SOP]).

1. INTRODUCTION

The Quality Assurance Plan (QAP) defines the organization and methodology that all partners shall apply throughout the project's lifecycle. It forms a common standard for the entire Project.

The QAP is intended to:

1. Define quality objectives.
2. Define roles and responsibilities to ensure successful progress of the Project.
3. Present the project organization, monitoring and reporting.
4. Describe software management practices: procedures and rules for the CrossGrid project.

As defined in [BRAUDE], QA consists of the following steps:

- specify how to manage documents,
- identify process,
- plan,
- design and build software,
- deliver and maintain the product.

These steps will be reflected in separate chapters of this Deliverable. For each step, a number of conformance tests exist and they will be presented as needed. [BRAUDE] also presents a standardized Quality Assurance Plan template (IEEE 730-1989 - [IEEE]) which is followed in substance by this document.

This deliverable is a result of the reevaluation of CrossGrid Quality Assurance needs and bases on the experience of CrossGrid partners, expert advice and information provided by other, associated Grid projects, such as DataGrid and GridLab.

2. QUALITY OBJECTIVES

The CrossGrid quality objectives are to ensure conformity to this Quality Assurance Plan and to provide quality support to partners, with respect to the main project characteristics.

The CrossGrid project is characterized by the following:

- long development cycle (36 months),
- distributed development of software (21 partners),
- evaluation and production phases that have to be managed in parallel with refinement phases,
- open source,
- modularity of layers and components,
- large-scale integration.

The size and scope of the CrossGrid project necessitates the use of a thorough and complete Quality Assurance Plan, which is described in the following chapters. The procedures related to CrossGrid QA can be divided into the following aspects:

- Documentation QA (chapter 4),
- Conventions and metrics QA (chapter 5),
- Software development QA (chapter 6),
- Testbed development QA (chapter 6),
- Management QA (chapter 6).

It is with respect to these goals that CrossGrid defines the following quality objectives:

- All deliverables are to be handled in on time and subjected to a review process, so that their contents can be found satisfactory both by Project Management and by the European Commission which is entitled to review CrossGrid documentation.
- All CrossGrid Partners should follow the same set of conventions and metrics, which adhere to industry standards regarding good design and coding practices, as well as reviews and internal audits of the generated code.
- The code produced by the CrossGrid technical tasks (i.e. Applications, Supporting Tools and Grid Services) should be handed in on time (as specified by the release workplan issued by Project Management) and perform as specified by their design (which is also subject to approval).
- CrossGrid testbeds are expected to be set up in time for the first software release and to perform at an appropriate rate of success (expressed by a separate quality indicator - see section 5.4).
- All reviews are to be conducted at the scheduled times and their results are expected to be satisfactory, as defined in the objectives of each review.

The Quality Assurance Plan is supplemented by the following documents:

- CrossGrid Technical Annex 1 (see [ANNEX]),
- CrossGrid standard operating procedures (maintained by TAT - see [SOP]),
- Deliverable templates for Software Requirement Specifications, Design Specifications, Prototype Specifications and Prototype Evaluations (maintained by TAT - see [TATWEB]),
- Document Naming Conventions (maintained by TAT - see [SOP]),
- Project Document Repository (maintained by CG Office - see [CGPORTAL]).

3. PROJECT PRESENTATION

3.1. PROJECT OBJECTIVES AND GOALS

The primary objective of this Project is to further extend the Grid environment to a new category of applications of great practical importance, and into 11 new European countries.

The applications we are interested in are characterized by the interaction with a person in a processing loop. They require a response from the computer system to an action by the person in different time scales; from real through intermediate to long time, and they are simultaneously compute- as well as data-intensive. Examples of these applications are: interactive simulation and visualisation for surgical procedures, flooding crisis team decision support systems, distributed data analysis in high-energy physics, air pollution combined with weather forecasting. A visualisation engine, used to automatically interface with external equipment, such as medical scanners or weather stations and convert the data to a predefined visual format, should be developed and optimised for these applications.

To enable efficient development of this category of applications for the Grid environment, new tools for verification of parallel source code, performance prediction, performance evaluation and monitoring are needed. This, in turn, requires extension of the Grid by new components for application-performance monitoring, efficient distributed data access, and specific resource management. Users should be able to run their applications on the Grid in an easy and transparent way, without needing to know details of the Grid structure and operation. CrossGrid will develop user-friendly portals and mobile personalised environments and will integrate new components into the Grid and application development tools.

The elaborated methodology, generic application architecture, programming environment, and new Grid services will be validated and tested thoroughly on the CrossGrid testbeds. This will result in further extension of the Grid across Europe. In our CrossGrid development we will exploit all the available achievements of DataGrid, EuroGrid and other related projects in a way which enables their interoperability. CrossGrid closely collaborates with DataGrid.

3.2. PROJECT STRUCTURE AND ITS RELATIONS TO QUALITY ASSURANCE

As mentioned before, CrossGrid is being developed by 21 separate entities from 11 European countries. Thus, CrossGrid is large as Grid-related European projects go and it requires extensive effort on the part of Project Management to maintain coordination between individual entities.

Most of the partners involved in CrossGrid are scientific research institutions, unused to the formal Quality Assurance methodologies and procedures enforced in major commercial enterprises (indeed, only two partners - Datamat from Rome and Algosystems from Athens are commercial in nature). Due to the diversity of backgrounds and modes of operation exhibited by the Project partners it is unfeasible to introduce a standard, formalized QA process. Instead, CrossGrid uses a custom system, based on the experience of fellow Grid projects and the requirements posed by its own Technical Annex ([ANNEX]).

Ideally, QA should be performed by a separate organization, but for CrossGrid purposes that duty will fall to Project Management (WP5), even though some subcontracting may be done with regard to unit testing (at the discretion of the Project Coordinator). Five Quality Assurance Officers will therefore be

assigned to each Work Package, with the WP5 QAO as the acting supervisor of this team (see Section 3.4.7 for an explanation of the duties of CrossGrid Quality Assurance Officers).

The following sections present the structure of the CrossGrid project.

3.2.1. Work Package WP1: CrossGrid Application Development

The main objective of this work package is to design and develop large-scale Grid-enabled applications for simulation and visualization that require real-time responses from the system. The main challenges are the distribution of source data, simulation and visualization tasks, virtual time management, simulation/visualization rollback due to user actions, and platform-independent VR visualization. This work package provides a representative collection of sample applications from various fields that can exploit the specific interactive functionalities developed in the CrossGrid project. Their need for specific services and their performance characteristics provides a driving force for other software development-related work packages. The applications will serve as benchmarks for CrossGrid performance, and crucially, they will demonstrate within their respective fields the added value provided by Grids in general, and technology developed by the CrossGrid project in particular.

Work Package 1 is expected to adhere to the following QA procedures (described in subsequent chapters):

- deliverable naming conventions, structure and submission procedure,
- code-related conventions and documentation requirements,
- usage of selected development tools,
- release preparation plans,
- publication policies,
- unit testing guidelines,
- managerial review requirements.

3.2.2. Work Package WP2: Application Development Support

For end users and program developers, one of the critical aspects of the Grid is its complexity. The development, debugging and tuning of parallel and distributed applications on the Grid is difficult and tedious due to the fluid and heterogeneous nature of the Grid infrastructure, which incorporates numerous hardware and software components. The aim of this work package is to specify, develop, integrate and test tools that facilitate the development and tuning of parallel distributed, compute- and data-intensive, interactive applications on the Grid. Verifying that user applications comply with the MPI standard will reduce the need for debugging sessions on the inhomogeneous Grid environment. In order to make parallel applications portable and reproducible, this work package develops a debugging and verification tool for MPI programs. Efficiently using the Grid as an environment for large-scale interactive applications requires the end-users to monitor and analyze the performance of their jobs. To this end, tools are being developed that automatically extract high-level performance properties of Grid applications. In addition, online performance monitoring tools are being developed to analyze performance data for detecting bottlenecks within applications.

Custom-tailored benchmarks are being prepared with regard to data transfer, synchronization, I/O delay and CPU utilization. Taken together with the monitoring tools, they will allow analysis of the system at any level of granularity, from the Grid level down to the process level.

The tools developed in this work package will be integrated into the testbed and will be promoted by and tested with the real end-user applications of WP1.

Work Package 2 is expected to adhere to the following QA procedures (described in subsequent chapters):

- deliverable naming conventions, structure and submission procedure,
- code-related conventions and documentation requirements,
- usage of selected development tools,
- release preparation plans,
- publication policies,
- unit testing guidelines,
- managerial review requirements.

3.2.3. Work Package WP3: New Grid Services and Tools

As a lot of Grid services and tools are already available and ready to be used, we only focus on those which are necessary for development of interactive compute- and data-intensive applications, and which have not been addressed in other Grid projects, as well as on services necessary for end users.

One of the portal features that we are implementing is roaming access, i.e. a mobile personalized environment. Secondly, we are addressing the construction of new resource management techniques based on self-adaptive scheduling agents for scheduling particular parallel applications submitted to the Grid. The goal is to achieve a reasonable trade-off between resource usage efficiency and application speedup, according to the user's optimization preferences. As our third objective, we are developing a prototype infrastructure for the needs of monitoring-related activities for automatic extraction of high-level performance properties and for tool support for performance analysis in work package 2.

Work Package 3 is expected to adhere to the following QA procedures (described in subsequent chapters):

- deliverable naming conventions, structure and submission procedure,
- code-related conventions and documentation requirements,
- usage of selected development tools,
- release preparation plans,

-
- publication policies,
 - unit testing guidelines,
 - managerial review requirements.

3.2.4. Work Package WP4: International Testbed Organization

The role of the international testbed is to assure the integration of all the applications, programming tools and new Grid services developed in this Project. Organizational issues, performance and security aspects, including network support, can only be evaluated with a testbed relying on a high-performance network (which is provided by the Géant project), with participation of an adequate number of computing and data resources distributed around Europe. The main objectives of the testbed organization are to:

- provide distributed computer resources where the Project developments can be tested in a Grid environment,
- integrate national Grids,
- integrate software developed within the Project,
- co-ordinate software releases,
- assure interoperability with other Grids, in particular with DataGrid,
- support the construction of Grid testbed sites across Europe.

Testbed sites are placed in 15 different institutions distributed across 9 different European countries, significantly expanding the Grid community.

Work Package 4 is expected to adhere to the following QA procedures (described in subsequent chapters):

- deliverable naming conventions, structure and submission procedure,
- usage of selected development tools,
- release preparation plans,
- publication policies,
- integration testing guidelines,
- managerial review requirements.

3.2.5. Work Package WP5: Project Management

This work package is devoted entirely to coordination, management and dissemination of project results. It develops no software but instead is aimed at managing other work packages. Within work package 5 there are such entities as the Project Office, responsible for project administration, the CrossGrid Architecture Team, devoted to developing the overall architecture of CrossGrid and its

standard operating procedures, the Technical Board, which has an advisory role in technical issues, the Internal Review Board, the role of which is to review project documentation, Central Dissemination, which publishes information regarding the project and promotes it on an international scale, and the Project Steering Group which oversees the organizational issues pertaining to CrossGrid.

Work Package 5 is expected to adhere to the following QA procedures (described in subsequent chapters):

- deliverable naming conventions, structure and submission procedure,
- publication policies,
- managerial review requirements.

3.3. PROJECT EVOLUTION

CrossGrid is scheduled for three years, officially starting in March 2002 and lasting until February 2005. The Project is divided into five distinct phases, as depicted by Fig. 3.1.

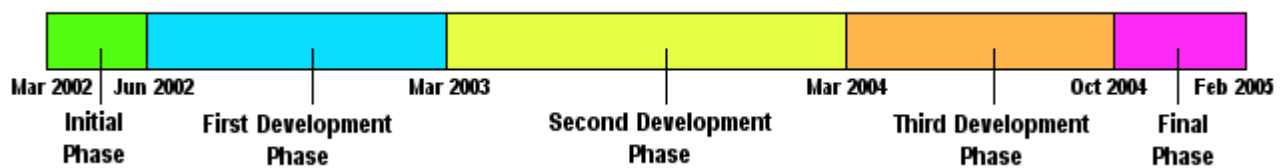


Figure 3.1: Project timeline

CrossGrid utilizes an incremental software release policy, with three scheduled releases. Therefore, the timeline is arranged as follows:

- Initial phase, including requirement definition and merging (Months 1-3),

Deliverable for all partners is D5.16 (report) at the end of M3. This phase merges all requirements (from WP1, WP2, and WP3) and specifies the structure of the CrossGrid software. This is done in close collaboration with managers and persons responsible for software integration.
- First development phase, including detailed design, refinement of requirements and production of initial prototypes (Months 4-12),

Deliverables for all partners are D5.17 (report) at the end of M6, and D5.19 (report) at M12. As the result, the first full specification of the CrossGrid architecture as well as operational procedures and the structure of deliverables are made available to all partners.
- Second development phase, including further refinement of prototypes and integration of components (Months 13-24),

In this phase there will be incremental improvement of specifications of the previous phase basing on the feedback from WPs. Deliverable for all partners are D5.19 (report) at the end of M15 and D5.20 (report) at the end of M24. The complete specification of the CrossGrid architecture will also be defined during this phase.

- Third development phase, including finalization of software components and their complete integration (Months 25-32),

Deliverable for all partners is D5.21 (report) at the end of M33.

- Final phase, including public demonstrations and preparation of final documentation (Months 33-36).

Documentation of CrossGrid architecture in the final version.

Each phase is set to produce different artefacts, expressed in the form of deliverables, and each will require different approaches to quality assurance. In addition, each development phase will conclude with the delivery of a new *software release*, as dictated by the provisions set out in [SOP] and in this document. The QA procedures applicable to each phase of the Project are fully described in chapter 6.

3.4. PROJECT MANAGEMENT STRUCTURE

The CrossGrid project is large, as the Grid idea calls for broad dimensions and integration, and it requires close cooperation of many institutions on the international scale. Such large projects are not unusual these days, and the experience with multi-national collaborations in research is very positive. They do, however, require extensive managerial effort in order to:

- ensure that each partner performs the duties assigned to him/her by the Project's technical annex,
- ensure that all deliverables are handed in on time and that each partner has sufficient resources,
- detect and help resolve problems which may arise due to insufficient communication between partners,
- coordinate the deployment of applications and testbeds (this work requires multiple Project partners working together).

It is therefore essential from the QA point of view that the Project Management Work Package be well staffed and competent in its duties, but it also proves beneficial to delegate some responsibilities to individual partners. The following sections describe the management structure, as applied to CrossGrid.

3.4.1. Managing of Work Packages

The CrossGrid project is divided into four Work Packages which deal with the technical aspects of the project, and one Work Package dealing with management, dissemination, and exploitation. All the technical tasks are coupled and they form one logical structure, with applications at one end and the infrastructure at the other, and with intermediate programming environment and software tools in-between. The whole project is factorised into a number of individual tasks, each of which has one or a few institutions attached. The technical and managerial responsibilities for the four Work Packages lie

primarily with their leaders, who are the top experts in their fields, and who originate from the principal contractors of the Consortium. It is up to these managers to prepare the work plan for these Work Packages and to report periodically on the progress. All these tasks have to be synchronised by the central management.

Each Work Package is coordinated by an appointed person, known as a Work Package Manager. That person is responsible for overseeing all aspect of the Work Package they're assigned to and reports directly to project management (see section 3.4.2). In addition, each task is managed by a single Task Leader, who performs the same function regarding his/her task (and is a subordinate of the appropriate Work Package Manager).

3.4.2. Central management and coordination

Figure 3.2 depicts the central management of CrossGrid.

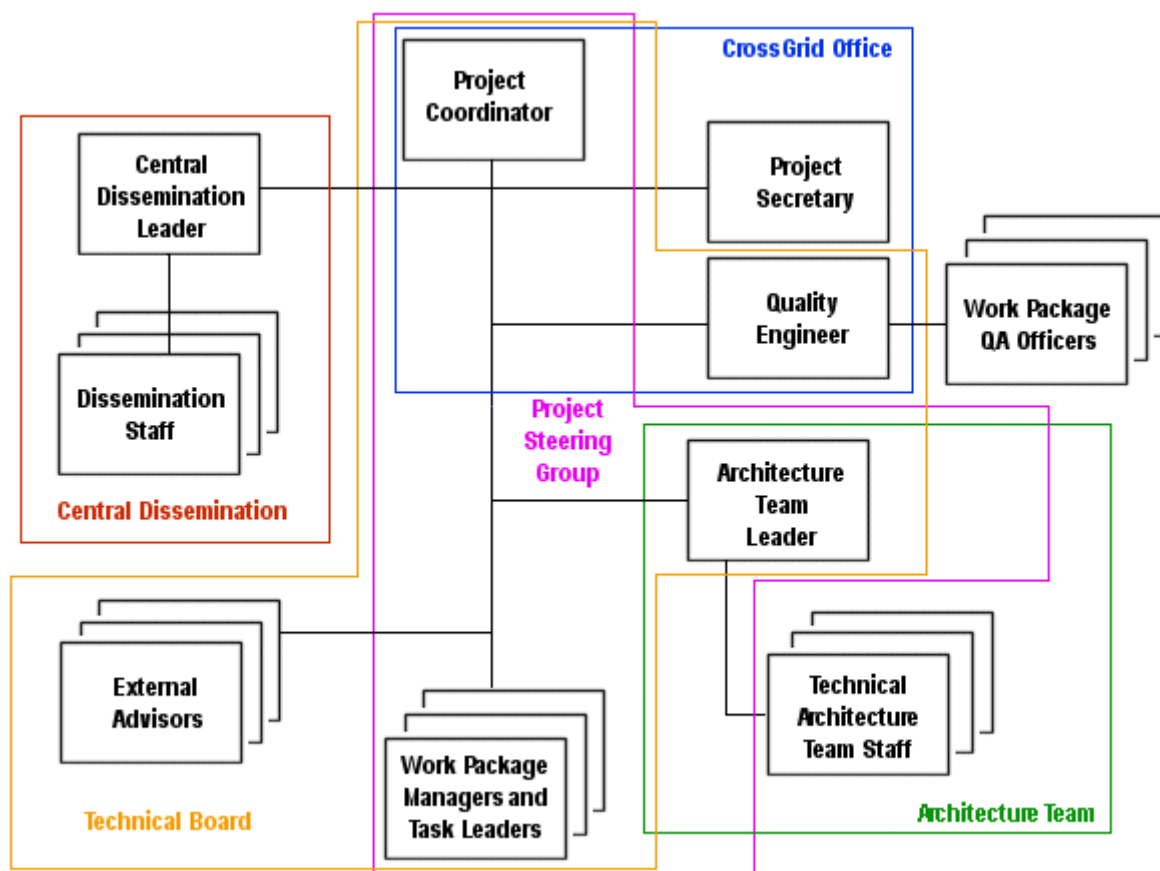


Figure 3.2: CrossGrid central management

The Project Coordinator, accompanied by the Project Secretary and the Quality Engineer, is responsible for:

- daily and long-term administration of the project,
- contacts with the IST Brussels office,
- organisation and running of the Steering Group meetings,

-
- contacts with partners for the purposes of conflict solving, and reporting,
 - supervising the execution of CG Quality Assurance procedures,
 - staff supervision.

The CrossGrid Office deals with all the administrative and financial issues. It is relatively small (three part-time personnel), but it is nonetheless necessary, as the CrossGrid Consortium is large, and to follow all correspondence, prepare monthly, quarterly and yearly reports, and to keep track of all these and many other documents requires quite an effort.

The role of the Architecture Team is crucial for the project. AT tasks are to:

- define the overall CrossGrid architecture,
- establish rules for component transfer from other Grids and integration into CrossGrid,
- select appropriate software engineering development methodologies,
- define measures necessary for interoperability with other Grids,
- define quality assurance criteria and procedures,
- review the technical progress of the Project.

The Architecture Team consists of an Architecture Team Leader, selected participants of the Project and invited experts. A subset of the Architecture Team is the Technical Architecture Team, which handles technical issues relative to CrossGrid software development.

The role of Central Dissemination will be discussed separately (see section 3.4.6).

The coordinating institution, Cyfronet, has a long record of providing good computational services to the research communities, and is one of the strongest partners in the Consortium. On a daily basis a scientist with good managerial experience on the international scale coordinates the project.

3.4.3. Steering Group

The technical and managerial matters are handled by the *CrossGrid Steering Group (SG)*, consisting of the Project Coordinator (PC), the leaders of the Work Packages (WPL), the leader of the Architecture Team (ATL), the leader of the Internal Review Board (IRBL), the Central Dissemination Leader (CDL) and a few experts. This group is also responsible for:

- defining the overall Project strategy and workplan,
- arbitration of both technical and policy-related conflicts (if arising),
- checking the monthly progress of the Project, measured by the milestones and deliverables, and taking actions when necessary,
- planning and preparing events concerning the whole consortium, workshops, conferences etc.
- coordinating the presentation of Project results.

The Steering Group can organise internal technical review panels whenever necessary. The group meets regularly, either in person or via phone/video-conference; meetings occur approximately every second week or at the request of Project Management. Minutes of these meetings and documentation of decisions and actions are available on CrossGrid websites.

3.4.4. Collaboration Board

The *Collaboration Board (CB)*, consisting of the representatives of each partner and chaired by a person elected at the beginning of the project, supervises the global progress of the project and deals with strategic and contractual issues concerning the project. This body is also in charge of resolving conflicts in the Collaboration. It meets yearly, but can also take action whenever necessary.

In a large and broadly distributed collaboration the access to information and the communication is essential. Collaboration Web pages are set up and maintained by one of the partners over the life of the project. A repository is created to keep all the documentation and publications (see section 4.4). E-mails will be the main communication tool for daily operations; operational meetings are conducted via phone and/or video-conference (VRVS). Annual workshops are held, at which the main results are presented, giving a basis for the annual reports.

3.4.5. Technical Board

The CrossGrid Technical Board, composed of the Project Coordinator, the Quality Officer, the Architecture Team Leader, Work Package Managers, selected Task Leaders and outside experts is an advisory body, with the following duties:

- review, analysis and evaluation of technical progress,
- assessment of quality procedures,
- identification of possible technical problems,
- recommendations to SG about directions of the future development as well as guidance on Project detailed scope and goals,
- maintaining contacts with other Grid projects.

The Technical Board will meet on a quarterly basis. In addition, each year, at the CrossGrid integration meeting it will give strategic advices based on the results of internal evaluation of the technical progress, recommendations about coordination of technical development.

3.4.6. Central Dissemination

Dissemination is a duty of every CrossGrid partner, and specific plans for each of them are presented in Chapter 9.3 of [ANNEX]. It is expected that representatives of every institution will spread information concerning the Grid via lecturing, seminars and contacts with local media, promote local developments on a broad scale on conferences and workshops, and writing scientific notes and publications, and play a visible role in the global development of the Grid by active participation in appropriate working groups.

However the project also requires a centralized effort to:

- design and maintain CrossGrid Web pages, which include a lot of practical information (address and mailing lists, conferences) for the whole life of the project,
- collect information on ongoing internal developments, and to make them available in an appropriate form for internal and/or public use,
- identify the CrossGrid products which could be of a broader interest, and to look actively for potential customers (*active promotion*),

-
- prepare CrossGrid products, including the documentation, in a form which could be accepted by customers, help in technology transfer, provide necessary advising and support,
 - prepare high quality PR materials.

Therefore the managerial structure of the CrossGrid is complemented with the central dissemination task, led by a Central Dissemination Leader, under the responsibility of Algosystems, who is an industrial partner, with experience in industrialisation, active promotion and public relations.

3.4.7. QA Officers

The exact structure of the CrossGrid project is further described in [ANNEX], but from the quality assurance viewpoint it is important to note that each Work Package employs a Quality Assurance Officer, responsible for day-to-day QA tasks within that Work Package.

In addition, the Project employs a part-time Quality Engineer, whose task is to directly coordinate and control all activities related to quality of software and other artefacts, coordinate the work of WP Quality Officers and work in close collaboration with the Project Coordinator, Work Package Managers and persons leading software integration tasks in all Work Packages:

- For WP1: Alfredo Tirado-Ramos
- For WP2: Roland Wismüller
- For WP3: Santiago Gonzalez
- For WP4: Rafael Marco

The Project Quality Engineer (QE) will also be responsible for supervising monthly unit tests and deriving the appropriate Quality Indicators (see sections 5.4.2 and 6.4). With respect to these issues, co-operation with a consulting company providing project management and QA services is also planned.

The current status of CrossGrid QA officers (as of May 2003) is as follows:

- For WP1: Elena Zudilova
- For WP2: Włodzimierz Funika
- For WP3: Paweł Wolniewicz
- For WP4: David Rodriguez
- For WP5: Piotr Nowakowski

Quality Assurance Officers (QAOs) are responsible for enforcing the conditions set out in this document and also in other documents which constitute the CrossGrid technical documentation (most notably the Standard Operating Procedures [SOP] document), for ensuring the quality of deliverables, gathering metrics, issuing progress reports (as described in the following sections) and for seeing that the proper process is followed in all aspects of the Project's technical lifecycle. Any shortcomings identified by the QAOs in this process will be pointed out to officials responsible for preparing each document and/or piece of code so that appropriate corrections can be made before that work is submitted. In particular, the QAOs will ensure that:

- all documents and deliverables are prepared according to the rules set out in [SOP] and in this document,
- CrossGrid software follows the specified coding conventions defined in [SOP],
- smooth progress of the Project is maintained and exhibited through specific Quality Indicators (see section 5.4).

Quality Assurance Officers will report to the Project Coordinator through the Quality Engineer.

4. DOCUMENTATION AND PUBLICATIONS

This section describes the layout and procedures applicable to CrossGrid deliverables and other documents.

4.1. DOCUMENT STRUCTURE

All CrossGrid documents shall be subject to a common template, presented at the CrossGrid Technical Architecture Team website (<http://kinga.cyf-kr.edu.pl/~tat/>). This template is modelled on a similar template used by the EU DataGrid project (with which CrossGrid is closely aligned). The formal template requires, that - along with the document's contents - the following information be provided:

- the name of the document as stated in the CrossGrid Technical Annex [ANNEX] (for documents not expressly mentioned in the Annex a set of generic names is provided - see [SOP]),
- document's authors and their affiliation,
- a config ID for the document (see section on naming conventions - below),
- document publication date,
- document release level (public, partially public or confidential),
- history of changes (type of alterations introduced since the last version of the document - a very short description).

Each document contains a log, which registers changes that have been made to its structure since its creation. In addition, deliverable documents contain a delivery slip, which has to be filled out by:

- the person preparing the current (final) version of the document for delivery,
- the person responsible for verifying the contents of the document prior to delivery (typically this would be the QAO of the appropriate Work Package),
- the person responsible for approving the document (typically the Project Coordinator or a person nominated for that purpose by the Project Coordinator).

All documents shall have a name and code consistent with the requirements of the CrossGrid Standard Operating Procedures. The following formalized grammar is to be applied when devising names and identifiers for new documents:

<Document name> = <project name><WP/task ID>-<document type>-<version>-<lead partner ID>-<description>.<extension>

<Config ID> = <project name>-<WP/task ID>-<document type>-<version>-<lead partner ID>-<description>

<Project name> = CG

<WP/task ID> = <WP number>.<task number> | <WP number>.<task number>.<subtask number> /* Requested by several task leaders */

<Lead Partner ID> = <boldface partner code><document number>

<boldface partner code> = CYF | ICM | INP | INS | UVA | SAS | LINZ | FZK | UST | TUM | PSNC | UCY | DAT | TCD | CSIC | UAB | USC | DEMO | AUT | LIP | ALGO

<document number> = <three-digit document counter value, specific to a particular partner>

<Document type> = D<deliverable ID> /*deliverable*/ | Min | Pub | Pres | Rep | Tem | Other

<Deliverable ID> = <WP number>.<task number>.<deliverable number>

<Version> = v<major revision number>.<minor revision number>

<Description> = <Short text describing the nature of the document (do NOT use spaces!)>

<Extension> = doc | tex | txt | ppt | xls | <other extension>

Please note that the **.doc** filename extension presented on the title page of a document does not necessarily need to change to **.pdf** if a conversion takes place.

There are additional examples and information provided in [SOP].

It is the Quality Assurance Officers' duty to inspect each document for conformance with the above-mentioned standards before the document is submitted for review and then delivered to the Project Office in Brussels.

4.2. DELIVERABLE REVIEW AND SUBMISSION PROCEDURE

CrossGrid employs a multilayered approach to deliverable review and submission. The Technical Annex [ANNEX] defines a list of deliverables, which have to be produced during the lifetime of the project. Figure 4.1 presents the QA plan is to be applied to each deliverable before it is delivered to the CrossGrid Office.

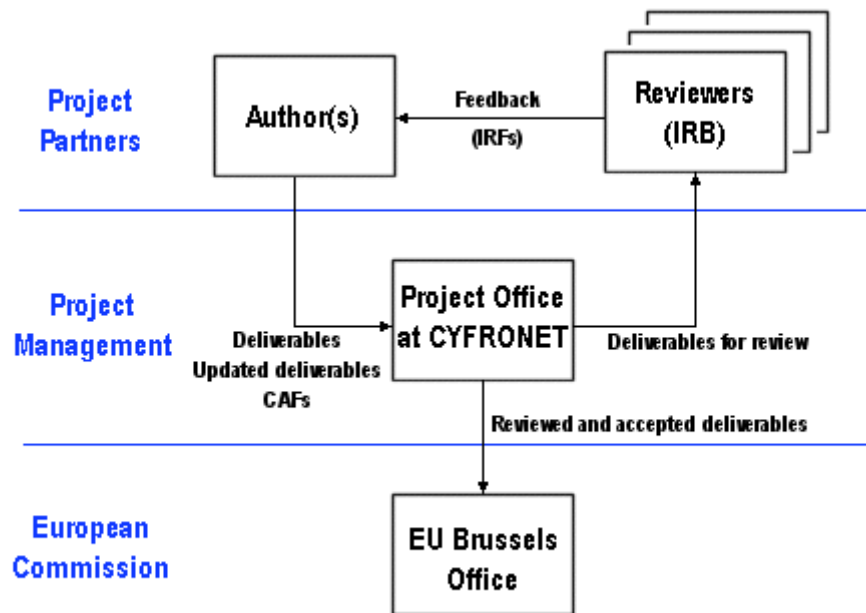


Figure 4.1: Deliverable review process

- The partner(s) responsible for the deliverable prepare a draft version approximately one month before the deliverable is due.
- The deliverable is subjected to internal inspection by the relevant Quality Assurance Officer (assigned to the Work Package which produced the deliverable). This is a technical inspection only (i.e. the document's matter is not reviewed at this stage).
- When the Quality Assurance Officer signs off on the deliverable, it is transferred to Project Management (WP5.1), which then contacts the CrossGrid Internal Review Board (IRB) [ANNEX] and requests a review.
- For each document one member of the Internal Review Board is selected as the *moderator*, responsible for coordinating the review process. The moderator assigns two or three reviewers to review the particular deliverable.
- The reviewers inspect the document's contents for conformance with the requirements of the Technical Annex. Each reviewer then issues a standard Internal Review Form (IRF), which details the results of the review and suggests changes that may be required before the final version of the document is submitted. The IRFs are handed back to the moderator with a recommendation on whether the deliverable should be accepted, accepted with changes or rejected. The moderator makes a final decision on the deliverable's acceptance or rejection (it should generally lie with the majority of reviewers) and inserts it into the IRF. The IRFs are then provided to the partner responsible for the deliverable being reviewed.
- When the review is complete, the document's author introduces the necessary changes to the document and fills out a standard Corrective Action Form (CAF), which lists all the alterations called for by the Internal Review Forms prepared earlier.
- The document is again submitted to Project Management for delivery to the CrossGrid Office and/or for release to the general public.

- If the document is not approved, the author needs to rework and resubmit it for a second internal review.

The Internal Review Board operates its own website linked to the central Project website. The address is www.es.uey.ac.cy/Projects/crossgrid/IRB/index.html. This site also details the internal review procedure and offers templates for IRF and CAF documents, along with other documents of note to CrossGrid reviewers.

4.3. CROSSGRID PUBLICATION POLICY

In addition to formal deliverables, which are internal in nature and mostly interest only CrossGrid partners and reviewers, the Project is also expected to generate a substantial amount of publications destined for groups not directly participating in the Project and for the general public. In light of this fact, CrossGrid Project Management has published guidelines for preparing and releasing such publications. Achieving uniformity of style and purpose is crucial to maintaining a positive image of CrossGrid on an international scale.

4.3.1. Types of publications

Publications may:

- be wholly or largely based on work done for CrossGrid (type a),
- contain some material based on work done in CrossGrid (type b).

4.3.2. Where to publish

Publications may appear in:

1. a refereed international journal or conference,
2. an editorial in an international journal,
3. as invited/keynote presentations at a conference,
4. a non-refereed international journal,
5. a Ph.D. thesis,
6. a refereed national journal or conference,
7. an internal report of an institute,
8. an M.Sc. thesis,
9. a colloquium,
10. a popular journal,
11. a newspaper interview,
12. a flyer, newsletter or web page.

This leads to publications of types **a1 - a12** and **b1 - b12**.

If a particular publication type is not included (e.g. conference poster not included in proceedings; stand at a fair), the author is expected to refer to a type with similar impact.

4.3.3. How to publish

All authors are expected to timely inform the consortium of the intention to publish, by sending a letter to the CrossGrid steering group (CG office) including an abstract of the publication. It is also essential to consult with other partners intending to publish at the same conference or in the same journal.

In addition to these general rules, there are also guidelines pertaining to specific types of publications. The authors are thus expected to:

For publication types a1 - a4, a6 and b1:

- include appropriate acknowledgement to CrossGrid, CrossGrid partners and the EC in paper or presentation.

For publication types a1, a6 and b1:

- inform CrossGrid partners about the publication and send an advance sample (if possible).

For publication types a12 and b1:

- after publication or presentation, submit an electronic or paper copy of the paper, or minimally the abstract to the CrossGrid Office for distribution among the partners and archival purposes.

For all types of publications:

- send the text of the publication to the CrossGrid Office at Cyfronet at the same time that the publication is submitted to the reviewed conference or journal.

4.3.4. Who can publish what

Basically, normal intellectual ownership and decency rules should be applied. Authors are free to publish their own results, within the constraints of the organization they're working for and barring objections from the CrossGrid steering group.

If the publication depends on someone else's results then the following rules apply:

- If the external results have been published in full in the open literature, the author should at least include a reference, but is strongly encouraged to consider the following two options.
- If the external results represent a major contribution to the publication, the author should obtain the originator's permission and make him/her a co-author.
- In all other cases, the author should seek the originator's permission, and the publication should include a clear acknowledgement and a "private communication" type reference.

4.3.5. Acknowledgements and standard phrases

As concerns acknowledgements to EU, the title page of the publication (or a separate acknowledgement section) should contain one of the following phrases:

- *The work described in this paper is/was supported by the European Union through the IST-2001-32243 project "CrossGrid".*
- *The work described in this paper is/was supported in part by the European Union through the IST-2001-32243 project "CrossGrid".*

When referring to contributions from other partners, the author is free to phrase the acknowledgement, but should include names, institutes and a reference to the CrossGrid project.

4.4. CROSSGRID DOCUMENT REPOSITORY

Access to all of current and past CrossGrid documentation for review and technical purposes is available at all times through the central Project website at www.eu-crossgrid.org. The Project section contains all deliverables (both submitted and pending) as well as the monthly reports from each partner (these contain per-task breakdown of work performed during each month and are later compiled into the standard Quarterly Reports - see section 6.7.2). Some deliverables are considered confidential and access to them requires authorization (the appropriate passwords are provided to project partners). Public deliverables are readily available to all concerned. This one-stop access to all the documentation generated by the project enables prompt location of important documents and facilitates the review process described in section 4.2.

The CrossGrid website also includes a separate page devoted to dissemination activities. This section is maintained by Algosystems.

The CrossGrid Document Repository is maintained by the CrossGrid Office at CYFRONET.

5. STANDARDS, PRACTICES, CONVENTIONS AND METRICS

CrossGrid is expected to deliver a substantial quantity of software, integrated and tested on numerous platforms. Owing to the complexity of the Project and the multitude of partners participating in it, it is essential that the same rules are enforced for each partner and each software module produced.

5.1. CROSSGRID STANDARD OPERATING PROCEDURES

A special document, mentioned before, details the standards, practices, conventions and metrics enforced within the CrossGrid project - namely the Standard Operating Procedures [SOP] document. In short, the document consists of the following sections:

- Central repository - information related to the functioning and terms of use of the CrossGrid code repository at FZK (Karlsruhe) and its backup site at Valencia. This repository is tasked with software collection, maintenance and retention.

The version control system chosen for CrossGrid is CVS; the interface is SourceForge.

- Tool control - a list of developer tools which should be used by all Partners. The selection of developer tools is proposed by the Technical Architecture Team in conjunction with individual Task leaders and then approved by Project Management.

In order to limit project costs, CrossGrid makes use of freeware and open source programming tools, where possible. Should a commercial tool require purchasing, an appropriate request should be submitted to the Project Coordinator, who is authorized to make decisions regarding software purchases.

- Problem reporting and change request mechanisms - all participants involved in developing code for CrossGrid are to follow the same procedures. The appropriate tools for problem reporting and change requesting are integrated into the CrossGrid code repository. They are consistent with the requirements of the Standard Operating Procedures document and they are intended to facilitate smooth development of the Project.

The CrossGrid issue handling system is integrated with the SourceForge CVS repository at FZK.

- The Autobuild tool - information relating to requirements posed by the Autobuild tool, used as a means of creating regular builds of CrossGrid software with minimum overhead for developers.
- CrossGrid naming conventions for software development - the largest section of the Standard Operating Procedures document is devoted to the procedures which should be followed in the implementation of CrossGrid modules. Naming conventions are discussed for:
 - packages
 - classes and interfaces

- methods
- constants, variables and enum types
- directories and files
- extensions
- comments
- special considerations for C, C++, Java, SQL and Perl programming
- Other conventions - such as date/time formats, measurement units etc.

5.2. SOFTWARE REQUIREMENT SPECIFICATIONS

The Software Requirement Specifications are a formalized way of expressing the desired functionality of the software being constructed. CrossGrid utilizes an SRS template modelled on the IEEE 830-1993 standard ([IEEE]) and adjusted to the requirements of CrossGrid technical tasks. The template is available at [TATWEB] and each partner is expected to fill it out in accordance to the requirements for its own software (see section 6.1).

SRS specifications are crucial for the development and maintenance of unit testing procedures, expressed as test case suites. See section 6.4 for details

5.3. DESIGN SPECIFICATIONS

Just like the SRS documents describe the Project in terms of software functionality, the Design Specifications are meant to be a technical description of the components required to attain that functionality. Therefore, prior to the implementation phase, each CrossGrid partner is expected to fill out an appropriate Design Template, modeled on the IEEE Software Design Document standard 1016-1987 (template available on [TATWEB]; see also [IEEE]). This document includes the detailed decomposition of each software module, expressed as UML class, sequence and component diagrams. See section 6.2 for more details.

5.4. QUALITY INDICATORS

CrossGrid will employ a system of quality indicators (QI) during its development phase. The CrossGrid Office will be responsible for compiling and publishing the current quality indicators each month. Three groups of quality indicators are envisioned, one for development process, one for code development (these will be task-specific) and one for testbeds. For each group, the relevant Tasks will be expected to submit monthly updates regarding their indicators. These reports should be submitted electronically to the Project Office. Basing on the information gathered, the CrossGrid Office will prepare detailed reports, including all the defined indicators and summary reports of the overall Project progress (basing on quality indicators). The main purpose of these report is to obtain a sense of how well the Project is progressing from the Quality Assurance viewpoint, without a need to analyze all indicators in detail. To this end, each indicator will be accompanied by a cutoff value, which - when reached - will trigger automatic mention in the reports. All the reports prepared by CG Office

will be published on the intranet pages (available from the main Project website) and accessible by authorized users only. The Project website will also contain report templates, prepared by the CrossGrid Office.

In general, the Work Package 5 Quality Assurance Officer will be responsible for ensuring process quality (Section 5.4.1), while the Project Quality Engineer will have responsibility for ensuring quality of code (Section 5.4.2 and also 6.4).

5.4.1. Process Quality Indicators

Process-related metrics are meant primarily to capture the current status of development and anticipate changes in resource requirements. The metrics proposed in this area will provide executive-level summary of the Project status. Each Project Partner will be responsible for creating and delivering monthly reports to the Work Package 5 Quality Assurance Officer no later than on the 5th day of each month (unless it happens to be Saturday, Sunday or a national holiday, in which case the first following workday will be the deadline). These reports will include the following indicators:

- number of person-hours allocated by each Project Partner to development of particular tasks,
- number of persons allocated by each Project Partner to development of all tasks during the measurement period,

The CrossGrid Office will be obliged to prepare, before the 20th day of each month, a summary report detailing all the information and indicators submitted by each Partner and, additionally, the following global indicators:

- number of monthly reports delivered to CG Office on time,
- ratio of deliverables submitted on time/total number of deliverables submitted for each WP,
- ratio of deliverables approved by the EU/total number of deliverables submitted for review (measured yearly),
- total number of deliverables that did not pass internal reviews (i.e. performed by IRB) for each WP,
- ratio of total number of person-hours reported by Project Partners per task/total number of person-hours planned for the task in the Project's schedule (measured on a quarterly basis).

5.4.2. Code development Quality Indicators

Quality of code development has been divided into two groups of indicators, as per [JONES]:

- static source code metrics and quality indicators,
- progress and effectiveness of testing.

For the latter type of indicators the collection of tests results is crucial. It should be also noted that testing policy is one of the critical factors that have major influence on the overall quality of the delivered software.

Figure 5.1 presents the CrossGrid unit testing schema. Please note that this figure is duplicated in Section 6.4, which details the exact contents of test case suites and the monthly reports being compiled.

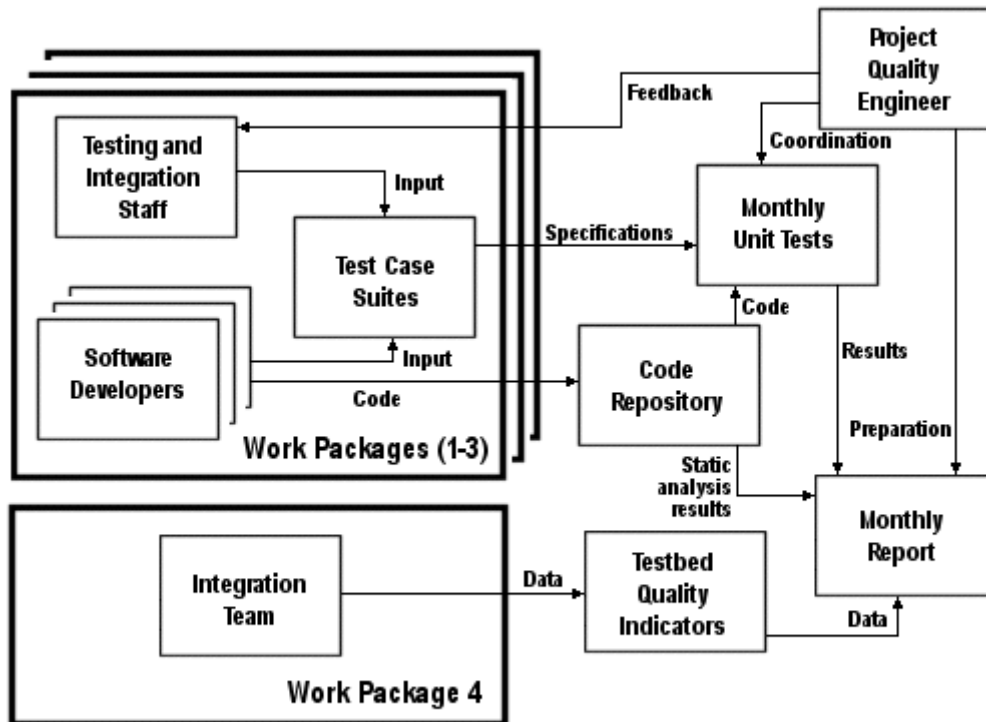


Figure 5.1: Software unit testing and report preparation

A thorough discussion of unit testing and their related quality indicators can be found in Section 6.4. In addition to the test metrics described there, the following factors related to software testing will be measured for each task on a monthly basis:

- amount of issues/defects reported in the project’s issues tracking system (using the Bugzilla tool as explained in [SOP]),
- average time required for issue resolution (measured and calculated for every WP),
- amount of pending (i.e. unassigned) issues,
- amount of issues with severity defined as *blocking*, *critical* or *major*.

At every monthly checkpoint the following criteria are also to be evaluated:

- there must not be any open and/or pending issues of *blocking* or *critical* severity,
- there should not be any open and/or pending issues of *major* severity,
- there must not be any open issues that were reported as open in the previous checkpoint report,
- deliverable code stability,
- overall test effectiveness.

The stability of the code is measured using indicators derived from metrics described above.

5.4.3. Testbed Quality Indicators

The following will be measured by the WP4 Quality Control Team (Task 4.4) and supplied to the Quality Officer each month:

- number of sites participating in the testbed,
- monthly uptime (for each testbed site),
- number of users (registered and active),
- job submittal success rate (number of successful jobs / total number of jobs).

All measurements will be performed on a monthly basis except where indicated otherwise.

6. TESTS, REVIEWS AND AUDITS IN THE CROSSGRID TIMELINE

In order for CrossGrid to develop consistent, well-documented software, a system of tests, reviews and audits must be put in place. This process is generally described in the Technical Annex [ANNEX], with particular points further clarified in subsequent deliverables, also foreseen by the Annex. As the Project progresses, numerous reviews and audits are conducted, depending on the nature of the phase which the Project is currently in:

- Initial Phase, including requirement definition and merging (Months 1-3),
- First Development Phase, including detailed design, refinement of requirements and production of initial prototypes (Months 4-12),
- Second Development Phase, including further refinement of prototypes and integration of components (Months 13-24),
- Third Development Phase, including finalization of software components and their complete integration (Months 25-32),
- Final Phase, including public demonstrations and preparation of final documentation (Months 33-36).

6.1. INITIAL PHASE: SOFTWARE REQUIREMENTS REVIEW

The Software Requirements Review is conducted upon the submission of the so-called Software Requirement Specifications (SRS) by each technical task (i.e. tasks, which develop software, as opposed to management tasks, integrated within WP5), as depicted in Figure 6.1.

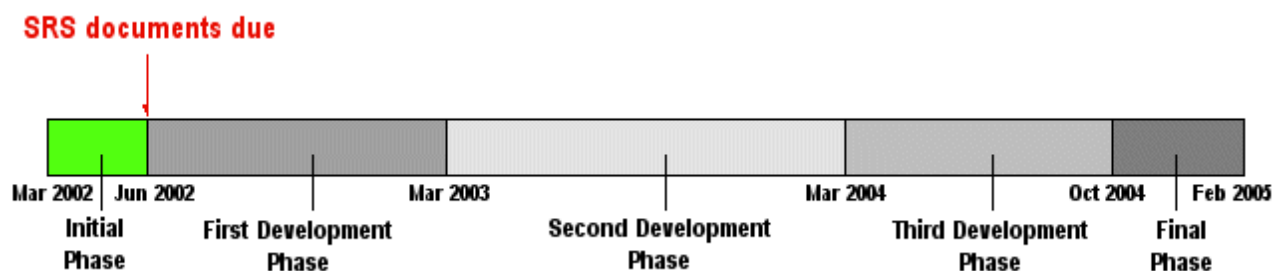


Figure 6.1: Software Requirements Review in the CrossGrid timeline

These documents contain the following information:

- overall description of module (functionality, operations, interfaces, requirements and constraints). Where appropriate, these should be augmented by UML component and use case diagrams,
- a “state of the art” document reviewing technologies suitable for the development of each module.

In addition, the Technical Architecture Team (TAT) is responsible for preparing a preliminary draft of the general project architecture, expressing any interaction between project components. Each task must fit into this architecture and the TAT is responsible for reviewing all SRSs and deciding whether they conform to that overall Project hierarchy. Upon completion of this review, the partners responsible for each SRS are expected to make corrections in accordance with the deliverable submission policy stated in Chapter 4.

A standard SRS template has been approved by Project Management and distributed to all relevant partners.

6.2. FIRST DEVELOPMENT PHASE: DESIGN REVIEW

Following the submission of SRS documents, each partner is to create a Design Document, expressing in concrete terms (UML class and sequence diagrams) the exact structure of their module (see figure 6.2).

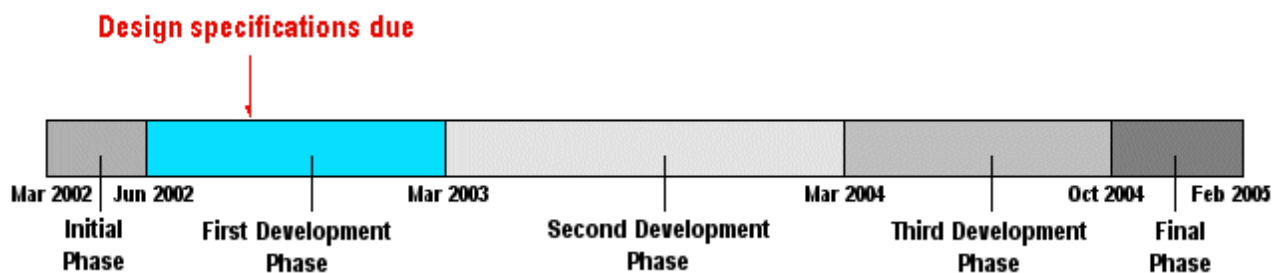


Figure 6.2: Design Specifications review in the CrossGrid timeline

The Design Document is intended as a “white box” description of document functionality and is subjected to review by the IRB and TAT. The Design Documents contain the following information:

- system decomposition description - an overview of the component breakdown of each module, accompanied by UML component diagrams. This section describes the module functionality in accordance with the IEEE 1016-1987 ([IEEE]) Standard Design Template,
- dependency description - this section is devoted to the interaction between the module being described and other modules (those being developed by other tasks as well as those external to the Project),
- interface description - a formal description of any interface the software will provide (both APIs and GUIs) including method declarations and descriptions along with use cases and sequence diagrams (screenshots should be provided for GUIs),
- detailed design - class diagrams for each component of the software module being developed. This is the most low-level description of software design, with explanations regarding each attribute and method of the software being developed.

The Design Documents are the basis for the implementation process, which commences upon the completion of their review.

A standard Design Document template has been approved by Project Management and distributed to all relevant partners.

6.3. ALL DEVELOPMENT PHASES: IMPLEMENTATION AND SOFTWARE RELEASES

As stated before, following the submission and acceptance of Design Documents, CrossGrid enters a phase of incremental releases, each of which is subject to a review. Three major releases are accounted for in the Technical Annex (see figure 6.3), but - as experience dictates - additional releases may prove necessary and the implementation process should be flexible enough to accommodate these releases. This issue has been addressed by the managers of the CrossGrid software repository, through the introduction of the Autobuild tool (see [GRIDPORTAL] for further details).

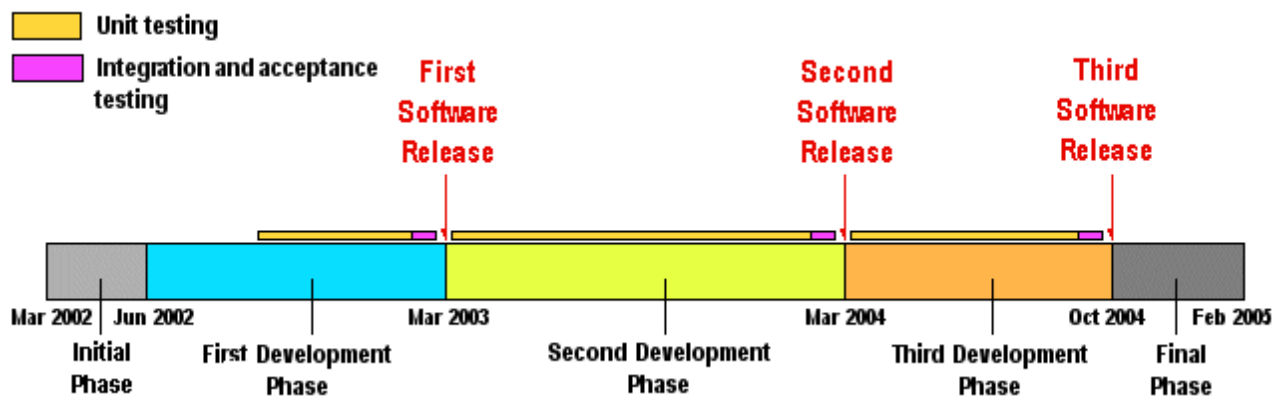


Figure 6.3: CrossGrid software releases

For each release, the following procedure is intended to occur:

- Coordination meeting*: involves a discussion of the previous release and develops a framework for a subsequent release.

The coordination meeting should be organized after each release and be open to all Project participants, but entail the participation of WP task leaders, CrossGrid steering group and representatives of the Technical Architecture Team and WP4 Integration Team.

The end product of this meeting should be an evaluation of the release's suitability and quality, along with a prioritized list of issues concerning the next release, including: all software and documentation modifications with regard to the previous release and versions of external software (i.e. Globus Toolkit, DataGrid middleware, compilers etc.) to be used for the next release.

In preparation for the coordination meetings it is expected that each task will hold internal meetings to develop a common stance and deliver a cohesive presentation.
- WP follow-up meetings*: individual tasks discuss the issues related to the next release and develop specific workplans.

Each task is expected to hold an internal meeting in order to develop its own workplan with regard to the next release, taking into account the results of the coordination meeting. The follow-up meetings should involve designations of responsibilities for individual participants.

A separate meeting should be held by the Architecture Team to gauge the influence of the coordination meeting on CrossGrid architecture. If significant changes are foreseen, an updated version of the architecture document must be developed.

- *Software release workplan coordination*: The CrossGrid Steering Group should develop a plan for the next release, upon discussion with individual WP leaders. The plan should include a list of all third-party software units which need to be installed on testbed sites in order to satisfy the upcoming release's requirements.
- *Middleware integration*: applications and middleware tasks upload the latest (internally tested) versions of their software to the central repository (Karlsruhe), where they are accessed by software developers for testing. Integration issues are reported and fixed through the use of mechanisms described in chapter 4.

During this process, the pertinent modules are integrated with external software units and basic integration tests are performed as per the software release test plan described in WP4 deliverables.

- *Acceptance testing*: an integrated CrossGrid release is installed on the Test and Validation testbed for acceptance testing performed by Task 4.4 (Verification and QA). The acceptance tests are deemed to have been passed when each scenario presented by the application groups is executed correctly.
- *Release*: The Integration Team holds a meeting to describe the release and indicate changes introduced since the last release. The release is accompanied by relevant documentation, developed by individual WPs and by the Integration Team.

The types of tests and the applicable procedures are further described in Appendix 4 of [TEST].

Figure 6.4 presents the release procedure diagram. As of April 2003, the initial Coordination Meeting has taken place in Santiago de Compostela (Spain) and the first prototype of tools from Work Packages 2 and 3 has been developed. Since Work Package 1 is on a different schedule, actual applications will not be integrated into the overall architecture before their delivery date (Month 18 of the Project's lifecycle).

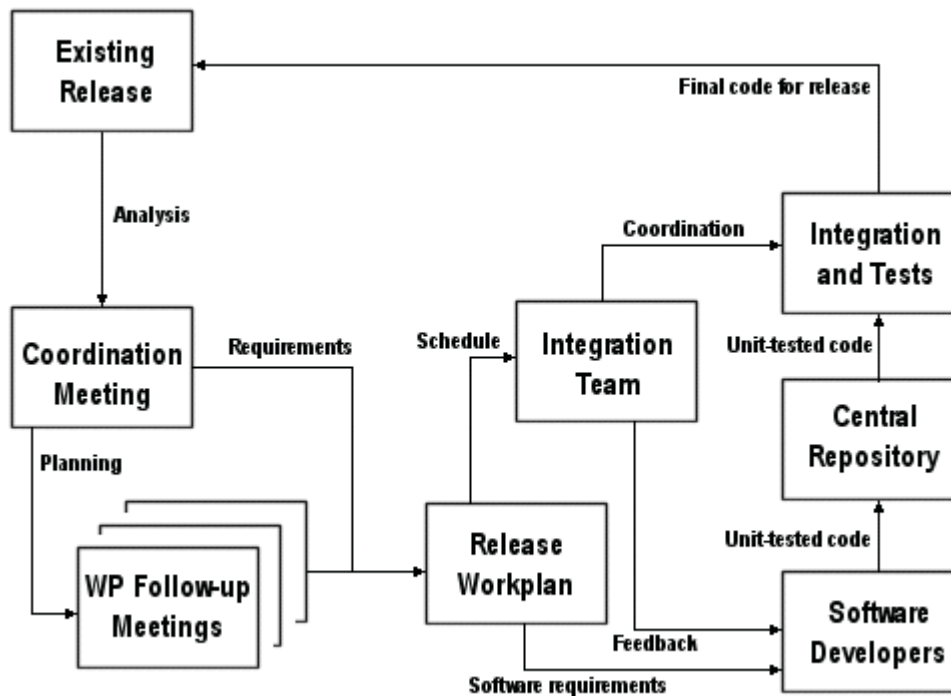


Figure 6.4: CrossGrid software release preparation procedure

For each major release, technical tasks are obliged to prepare a set of documents detailing the structure and functionality of the software they are delivering. The initial prototype documentation template has been approved by Project Management and distributed to all relevant partners. The template includes the following information:

- description of module components which have been implemented so far and the functionality of the prototype (with use cases),
- description of discrepancies between the structure of the prototype and the Design Documents submitted earlier. If any changes are introduced during the implementation phase, partners are expected to explain their rationality,
- instructions and requirements for operating the prototype (its installation and running),
- description of tests which the software has been subjected to (both “black box” - i.e. comparison of actual output with required output and “white box” - confirmation of expected behavior of individual components),
- any issues which may affect the functioning of the prototype, including any known bugs,
- references to properly annotated source code, contained in the CrossGrid repository at FZK (see above).

6.4. ALL DEVELOPMENT PHASES: UNIT TESTS

Unit tests are carried out for each task separately and proceed in a formalized manner, as specified in Section 5.4 (regarding Quality Indicators). Basing on the experience of multi-partner project

([BASSIN]), we will utilize, as one of the foremost Quality Assurance mechanisms, the evaluation of test results. Project management and Quality Assurance will both be enforced through collecting and publishing appropriate metrics and indicators, which - when appropriate - will direct the attention of Project Management to any potential problems and conflicts (on a monthly basis).

Figure 6.5 depicts the CrossGrid unit testing and report preparation procedure.

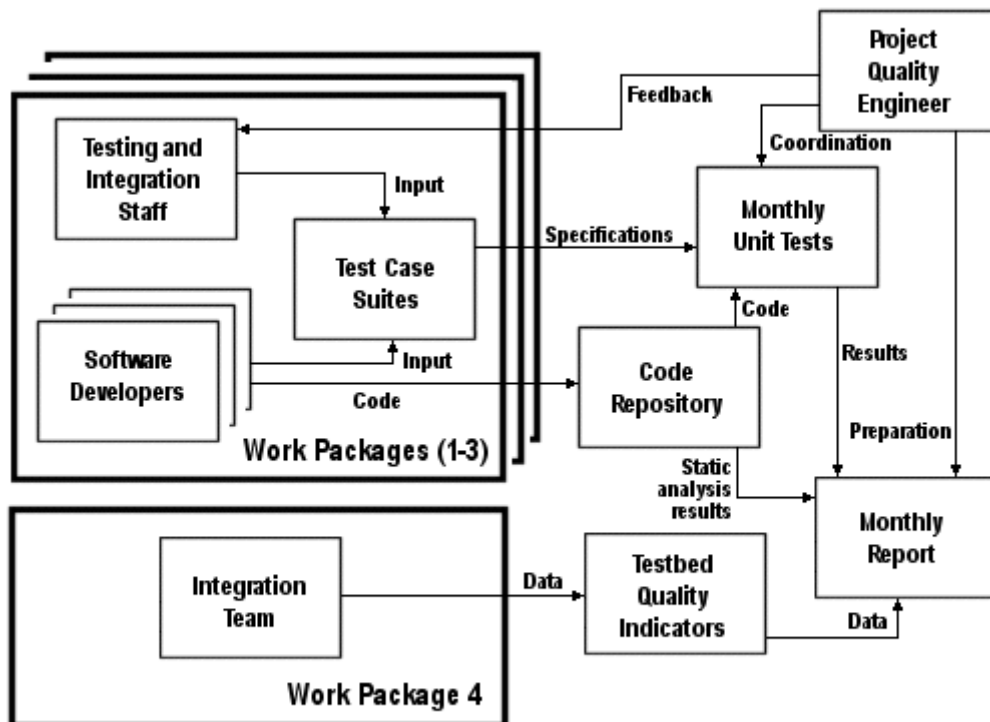


Figure 6.5: Software unit testing and report preparation

The first order of business is to prepare a test suite for each task which deals with software developments (this involves all the CrossGrid applications and most tasks from Work Packages 2 and 3). The relevant Integration and Testing tasks are required to prepare a set of *test cases* for their software. These are expected to be automated, i.e. they will use code uploaded to the central CrossGrid repository and provide a set of scripts so that each test may be run by persons not directly involved in the development of the package being tested. In order to achieve uniformity of test suites for each task, individual task leaders will be required to provide a formal description of their test suites, by filling out specially-prepared forms. These will include the following information:

- name of test case,
- description of the behavior being tested, with a reference to the appropriate SRS document,
- list of software components required to conduct the test (it is assumed that the test environment will be prepared by developers of the software module being tested),
- conditions for success,

- possible error codes (if the test is not successful) and their meaning.

Each test script will generate a simple output file, containing the word **OK** if the test is deemed to have been successful, or an appropriate error code (in case of failure).

The Integration and Testing tasks from each Work Package will be responsible for conducting the testing and for submitting to the Project Quality Engineer, no later than on the 10th day of each month (barring weekends and national holidays), a report including the following metrics:

- total number of system usage scenarios defined in SRS based on which the tested code had been developed,
- total number of test cases defined,
- total number of test cases executed,
- total number of test cases that are deemed to have been successful,
- total number of test cases that have failed,
- total number of defects reported for every test case,
- identifiers (test case class names) of test cases that have failed.

Using this information, the Project Quality Engineer will prepare a summary report, including the following indicators:

- ratio of total number of system usage scenarios defined in SRS/total number of test cases defined,
- ratio of total number of test cases defined/total number of test cases executed,
- total number of defects reported during execution of test cases,
- total number of defects reported per one test cases,
- total number of test cases that have passed,
- total number of test cases that have consistently failed,
- ratio of total number of test cases passed/total number of test cases executed,
- ratio of total number of test cases that have changed status from pass to fail.

On the last workday of each month, the Project Quality Engineer should also perform the following actions, meant to gauge static indicators, as mentioned above:

- set a tag on the entire source tree in the Project CVS repository,
- collect the tagged version of source files and compile it,
- in case of compilation errors, the QE should immediately inform the leader of the Task which produced the erroneous software. The deadline for resolving such issues is the second workday following the report,
- for each Work Package, perform a static analysis of code, including:
 - the amount of additional lines of code since the last checkpoint (uses CODECOUNT (<http://sunset.ucs.edu/research/CODECOUNT>) or RSM (<http://msquaredtechnologies.com/m2rsm/index.html>) for Java, C and C++ apps),

- gauging compliance with coding conventions defined in [SOP] (uses JWalk for Java apps (<http://www.crwydrwr.demon.co.uk/JWALK/>) and CodeWizard for C++ apps (www.parasoft.com/jsp/products/home.jsp?product=Wizard)),
- deriving the following metrics:
 - Cyclomatic Complexity Number (McCabe metric) (uses QStudio for Java apps (<http://www.qa-systems.com/products/qstudioforjava/index.html>) or RSM for Java, C and C++ apps),
 - count of packages, classes, functions and inner classes (uses JavaNCSS for Java apps (<http://www.kclee.com/clemens/java/javancss/>) or RSM for Java, C and C++ apps),
 - count of formal Javadoc comments per class and method (uses JavaNCSS for Java apps or RSM for Java, C and C++ apps),
 - quality rules (uses QStudio for Java apps or RSM for Java, C and C++ apps),

The monthly report will also include testbed Quality Indicators, described in Section 5.4 and supplied by the WP4 Integration Team.

The *ratio of total number of system usage scenarios defined/total number of test cases defined* is used as an indicator of the completeness of testing procedures defined in the Project. There must be at least one test case defined for each system usage scenario.

The *total number of defects reported during execution of test cases and ratio of total number of test cases defined/total number of test cases executed* are used as an indicator of progress relative to the completeness of the planned test effort.

The *ratio of total number of test cases passed/total number of test cases executed* is an important indicator of code quality and stability.

The *total number of test cases that have consistently failed* is an indicator of code quality and enables the identification of code that represents obstacles to progress through test activities.

The *ratio of total number test cases that have changed status from pass to fail* is an indicator that provides information about defects injections i.e. degree to which inadequate or incorrect code changes were being made.

6.5. ALL DEVELOPMENT PHASES: TESTBED RELEASES AND VALIDATION

A “testbed release” identifies a well defined and complete collection of software to be deployed on a well defined set of distributed computing resources, including the corresponding installation mechanisms.

6.5.1. Testbed release description

According to the CrossGrid Architecture [ARCH], the testbed supports the software components which are being developed within the framework of the project (CG software) and those from Globus and DataGrid (collectively referred as EDG software).

The CrossGrid testbed is divided into three focused testbeds, called the *development testbed*, *test and validation testbed* and *production testbed*, used at various stages of the software release process. The development testbed is relatively the smallest and supports day-to-day testing of software under development at various participating institutions. The test and validation testbed is used during integration meetings and the software release preparation procedure (see section 6.3), while the production testbed will be used for actual operation of the completed applications and tools. Figure 6.6 presents the testbed development timeline.

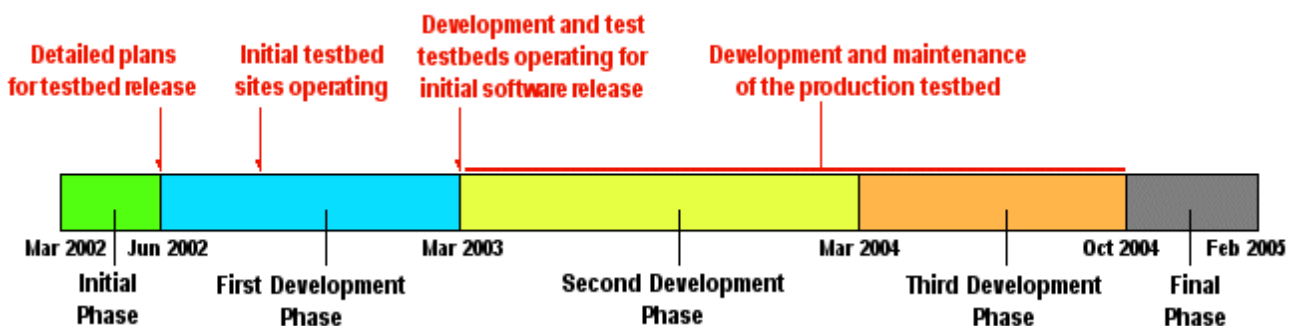


Figure 6.6: CrossGrid testbed development timeline

EDG release packages are designed to work with the LCFG installation tool. They are available from their official CVS repository. A similar scheme will be used in CrossGrid. The release includes the software and corresponding configuration to be used in each testbed resource: LCFG templates include the list of packages (RPMs for the RedHat environment), a configuration file that has to be customized at each site, with the corresponding instructions.

For convenience reasons, the full CrossGrid testbed release, including the EDG packages, is available from a single entry point at the code repository GridPortal hosted by FZK. The current production and test and validation testbed packages can be found there, and the development testbeds will be hosted also there when required.

The WP4 Integration Team is responsible for maintaining the CG middleware package, including any required external software. The Integration Team should also arbitrate potential conflicts resulting from the use of varying versions of the same software by different tasks.

Below the previous link, three different areas can be found for each release:

- the **cg** area includes the Crossgrid software,
- the **edg** area includes the copy of the corresponding EDG release,
- the **other** directory hosts the external software not included in the EDG release.

The current production testbed version (as of December 2002) has been labeled as 0.0.1. It corresponds to the EDG version 1.2.3. The current test and validation testbed version is 0.0.3, and corresponds to EDG version 1.4.2. The **edg** subdirectory is labeled after the EDG version numbering to avoid misunderstanding (i.e., `edg-v1_4_2`), but is unique for each version.

6.5.2. Formal steps in the preparation of a new testbed release

The preparation of a new testbed release is triggered by the need to incorporate new software either developed inside the project or externally.

We consider testbed releases aimed to reach the production testbed: releases in the development testbed can be as frequent as needed as long as the resources are available, while the test and validation testbed procedures have been described in detail in CrossGrid Deliverable D4.1 ([PLAN]).

The testbed release requires the agreement of the different WPs, and in particular of the WP4 integration team, the role of which is to ensure that all testbed sites use a uniform software package and to resolve any pending technical issues and conflicts, which may arise between various sites. Requests should be addressed in principle through the WP4 coordinator. Typical requests include the need to deploy new applications or middleware, keep EDG compatibility, fix serious bugs, etc.

6.5.2.1. New software: installing at the CVS repository at FZK

The first mandatory step for any new software to be deployed in the future in the CrossGrid testbed is its installation in the CVS repository hosted by the GridPortal at FZK. Details about its functionalities and use can be seen in Deliverable D4.3 ([STAT]). The software in the corresponding WP tasks directories will be copied to the distribution area when the release is ready.

External software not directly related to an application, tool or middleware, will be posted in the WP4 area. In particular the EDG software is copied by the WP 4.2 (Coordination with DataGrid) task people at FZK for each new EDG release.

6.5.2.2. Preparing the release proposal in the development testbed

The development testbed sites should download the new or modified software from the corresponding CVS repositories, and install it. Involving developers and, in what possible, WP4 people in this step should guarantee that modifications are incorporated to the software repository and also that RPMs are prepared for the software components, and also the corresponding configuration files for LCFG. The software with corresponding tests, and the documentation, should be posted and made available in the corresponding software repository.

6.5.2.3. Test and validation

As stated in the middleware test procedure in [PLAN], developers should transmit to the WP4 integration team new or updated middleware components, but also the WP4 integration team should identify them. This is in particular true for external packages. The test procedure itself is carried mainly by the WP4 task 4.4 in the test and validation testbed, and includes two different phases: in the alpha phase each delivered component is tested first “alone” and then integrated with other components and across test sites; in the beta phase knowledgeable users and other middleware developers are invited to perform tests using real applications.

6.5.2.4. Defining the testbed release components and calendar

Reaching a complete and coherent set of software components, tested and validated, that can be deployed in the production testbed is under the responsibility of the WP4 integration team. Too

frequent releases of the production testbed are not desirable. Usually they will take place in coordination with major software releases in WP1, WP2 or WP3. So the frequency will be in the range of 3-6 months. Releases in the test and validation testbed will be much more frequent.

6.6. ALL PHASES: PUBLICATION REVIEWS

The publication policy asks for an internal review of CrossGrid publications prior to their submissions. The CrossGrid steering group members will check the form and the content of publications, and for questionable cases internal experts will be assigned.

When the authors and the internal reviewers cannot come to agreement on a publication, they have to agree on an arbitrator whose decision is final.

When two partners disagree on a publication and cannot reach an agreement, they have to agree on an arbitrator whose decision is final.

6.7. ALL PHASES: MANAGERIAL REVIEWS

CrossGrid is subject to a number of managerial reviews. These include the twelve Quarterly Reports, scheduled every three months and three Yearly Project Reviews, conducted at the Project Office in Brussels.

Figure 6.7 presents the CrossGrid review timeline.



Figure 6.7: CrossGrid managerial review timeline

In addition, most CrossGrid quality indicators are computed on a monthly basis, along with descriptions of work submitted by each partner - their results are analysed by the CrossGrid Office on the fly.

6.7.1. Monthly Reports

In addition to compiling quality indicators described in section 5.4, each month the CrossGrid Office at CYFRONET is responsible for querying each partner for the work done during the reporting period. Partners are expected to submit detailed lists of actions performed (along with the time it took to perform each of them) to the CrossGrid Office. This data is reviewed internally by Project Management and any problems are discussed during Steering Group meetings. There are no formal artefacts of these monthly reports, but the information delivered to the CrossGrid Office is later included in Quarterly Reports (see next section).

6.7.2. Quarterly Reports

Once every three months, CrossGrid Project Management prepares a Quarterly Report of the Project's progress. This report is compiled with information obtained from individual partners and consists of the following sections:

- an executive summary of the Project's progress in the reporting period,
- work progress for each individual Task, including the status of deliverables and milestones (pursuant to [ANNEX]) and any issues reported by partners,
- a dissemination report, detailing the meetings, conferences, events, articles, publications etc. involving the Project,
- effort summaries for each partner (i.e. amount of person-months (PMs) assigned vs. amount consumed).

The Quarterly Reports are submitted to the Project Officer in Brussels, but they also aid Project Management in identifying and resolving any emerging issues related to the Project's organizational structure.

Figure 6.8 presents the Quarterly Report preparation process.

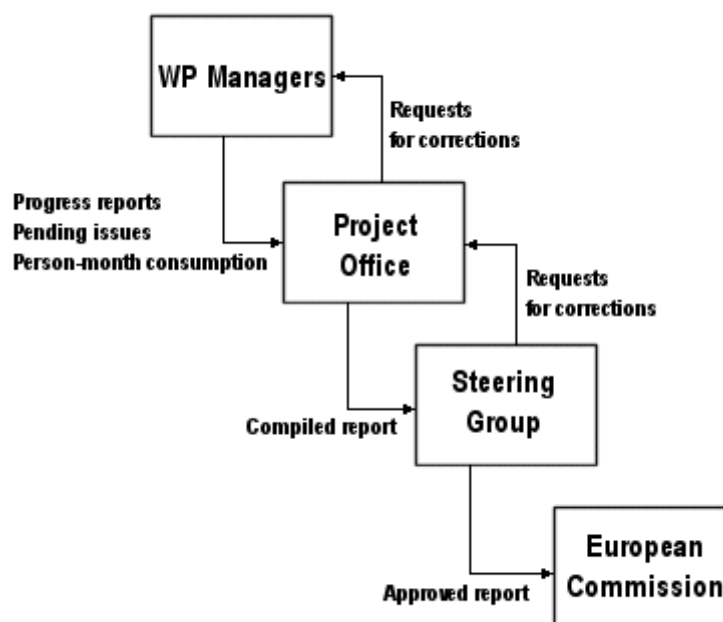


Figure 6.8: Quarterly report preparation procedure

The CrossGrid Office at CYFRONET is responsible for gathering all relevant information from each Project partner and compiling a Quarterly Report once every three months. This should take place before the end of each quarter, so that any pending issues may be resolved and the final document presented to the EC on time, along with other deliverables due that quarter.

Quarterly Reports are not reviewed by the IRB. Instead, they are submitted to the Project Steering Group, who can then identify any problems which warrant closer attention from Project Management (this is one of the key purposes of Quarterly Reports).

6.7.3. Yearly Reviews

In addition to the ongoing QA review process, provided for in the Technical Annex, the entire Project is subject to yearly reviews by the Project Office in Brussels. The procedure of the review is as follows:

- CrossGrid is expected to submit all its deliverables on time (as specified in [ANNEX]) to the Project Office in Brussels. The Project Officer is responsible for nominating external experts to review the contents of deliverables and point out any shortcomings.
- Once a year (after Month 12, Month 24 and Month 36), the Project Office conducts a review, which bases on the recommendations provided by the external reviewers mentioned above. Before the review takes place, it is expected that Project Management officials will deliver a presentation of the Project's results, point out any issues which affect the Project as a whole and answer all questions which may be posed by external reviewers. Each review is also preceded by an annual CrossGrid conference, the aim of which is to summarize the progress of the Project.
- Upon completion of the review, the Project Officer gives a final assessment of the Project's progress and makes a decision on whether to carry on or discontinue the Project.

7. CROSSGRID CORPORATE IDENTITY

In order to assert CrossGrid corporate identity, a set of document templates are provided and it is expected that all CrossGrid partners will - for all intents and purposes related to the CrossGrid project - use the available CrossGrid identity characteristics, namely:

- standardized document templates, both for deliverables and other documents,
- standardized transparency templates,
- standardized presentation templates (in PowerPoint),
- the CrossGrid logo,
- uniform color schemes for printed material and presentations,
- references to the European Union as sponsoring the Project (see section 4.3.5).

In addition, there exists an official CrossGrid presentation, which is regularly updated by CrossGrid Central Dissemination and meant as an overarching introduction to CrossGrid and all its aspects, presented in a widely-understood fashion. This presentation is freely available to the public and is expected to be used for official purposes.

8. CLOSING REMARKS

8.1. APPLICATION AREA

This Project Quality Plan shall be applied:

- by all partners,
- for all deliverables submitted to the European Commission,
- for all deliverables exchanged between partners,
- for all CrossGrid publications.

Consortium Partners supervise and check the work performed by their own staff in accordance with the CrossGrid QAP. This QAP is designed to be consistent with ISO 9001 and should be interpreted with reference to:

- terms and Conditions of EC contracts,
- the CrossGrid Technical Annex [ANNEX].

8.2. QUALITY ASSURANCE PLAN EVOLUTION PROCEDURE

Different events may necessitate modifications of the content of this document. Any project partner may request amendments but each amendment must be analysed by the appropriate Quality Assurance Officer and then approved by the CrossGrid Project Coordinator.

8.3. LACK OF ADHERENCE TO PROJECT QUALITY ASSURANCE

If there is a conflict between a Consortium partner's Quality Assurance procedures and those imposed on this Project, this should be brought to the attention of the Project Coordinator. In normal circumstances, precedence shall be given to this Quality Assurance Plan.