



QCDgrid Software Release Plan

Project Title: QCDgrid

Document Title: QCDgrid Software Release Plan

Document Identifier: QCDGRID2_RELEASE_PLAN_1.0

Document Filename: qcdgrid_release_plan.doc

Distribution Classification: Public

Authorship: Mark G. Beckett, Daragh J. Byrne

Approval List: QCDgrid Project Team

Distribution List: UKQCD Collaboration

Document History:

<i>Personnel</i>	<i>Date</i>	<i>Summary</i>	<i>Version</i>
MGB, DJB	30 th March 2005	First release version	1.0

Contents

1	Introduction	2
1.1	The QCDgrid Project	2
1.2	Scope and motivation for this document	2
2	Software Versioning	3
2.1	CVS Tagging	3
2.2	New software components.....	4
3	Problem resolution	5
3.1	Reporting a bug or issue.....	5
3.2	The lifecycle of a PR	5
4	QCDgrid release schedule.....	7
4.1	Emergency bug fixes.....	7
4.2	Initiation of the new versioning plan.....	7
5	Conclusions.....	8
6	References.....	9

1 Introduction

1.1 The QCDgrid Project

The UKQCD Collaboration aims to “procure and jointly exploit computing facilities for lattice field theory calculations, whose primary aim is to increase the predictive power of the Standard Model of elementary particle interactions through numerical simulation of Quantum Chromodynamics”. Such numerical simulations produce significant amounts of data in the form of binary files. The purpose of the QCDgrid project is to provide a software application and supporting infrastructure that simplifies the management, storage and manipulation of this data.

In the first three years of the project (2002 – 2004), software engineers at EPCC developed a software application called *QCDgrid* – a data management system that combines the distributed resources of the collaborators into a robust facility called the *UKQCD Grid*. The result is a multi-terabyte storage facility over six UK sites at: Edinburgh (including the University of Edinburgh Advanced Computing Facility), Liverpool, RAL, Southampton, and Swansea. Glasgow are also a member of the consortium.

The facility is based on commodity hardware and open-source software. The hardware consists primarily of high specification PC-based servers running the Linux operating system and managing large RAID storage arrays. On top of this infrastructure, the QCDgrid software (built with Globus Toolkit 2.4, EGEE, and an XML Database Server (XDS)) provides *Datagrid* management and user functionality – furnishing a simple and intuitive environment that hides the complexities of the underlying grid and presents a standard file system to the user. It incorporates a robustness metric that automatically disperses datasets across the grid, providing a resilience that ensures data is not affected by the loss of one (or possibly more) storage nodes.

QCDgrid allows the user to query and manipulate associated metadata using a *Metadata Catalogue Browser*. The software provides a *Job Submission System* that allows a user to schedule computations on remote HPC systems, from the comfort of their desktop computer. Security is leveraged from the Globus Toolkit, based on digital certificates issued by the UK e-Science Certificate Authority. The result is a reliable, secure data management system.

1.2 Scope and motivation for this document

This document details the framework under which updates to the QCDgrid software will be released during the second phase of the project. It also defines a formal process that addresses functionality requests, problem reports, and support issues related to the QCDgrid software and the UKQCD Grid.

During the first phase of the project, when the QCDgrid software was being developed to an operational level, releases occurred in an “ad-hoc” manner as the development team progressed through the project work plan. The latest version of the software source code was always available from the project NeSCForge CVS repository [2] and was typically deployed to the nodes of the UKQCD Grid either by a member of the project development team or local administrative staff at each particular site.

Problem reports and functionality requests were filed using a number of mechanisms, including: NeSCForge postings, e-mail sent directly to the developers, and feedback through the monthly UKQCD Software video conferences. No formal mechanism for prioritising bugs or features was applied and requests/reports were tackled as and when effort became available.

Now that the software has reached a stable and complete implementation, it is appropriate to define a formal release strategy, designed to streamline the future development of QCDgrid. In addition, a process should be defined to tackle bug/issue reports in an effective and efficient manner. To this end, the document defines two processes:

- **Process 1:– Software Versioning** describing how updates to the QCDgrid software application suite are to be published;
- **Process 2:– Problem Resolution** describing how problems with the QCDgrid software and issues with the UKQCD Grid infrastructure are to be tracked/assigned/resolved.

2 Software Versioning

With any software application, it is important that interested parties such as users, administrators and developers are able to uniquely identify each release of the application. For example:

- an administrator may need to confirm that they have installed a current and supported release of the software;
- a developer may need to attribute problems (bugs) to the specific releases of the software that they affect.

To provide such a mechanism, a formal versioning strategy shall be implemented. To facilitate this, the QCDgrid application suite will be regarded as consisting of two components:

1. **The Datagrid and Job Submission component** (both server-side and client-side elements) which:
 - a. supports the reliable storage and retrieval of QCD data and metadata;
 - b. allows the submission of one or more jobs to a compute node.
2. **The Metadata Browser component** which allows the metadata catalogue to be queried and related data to be processed/manipulated.

Versioning of the QCDgrid software will be done on a per-component basis: that is, the *Datagrid and Job Submission* component and the *Metadata Browser* component will each be versioned independently.

For each component, the versioning identifier “*Version X.Y.Z*” will be adopted, where X, Y, and Z are defined, as follows:

- *The Suite Version Number (X)* is the version number of the QCDgrid software suite – any release of the Datagrid and Job Submission component will be compatible with any version of the Metadata Browser component, provided that they share the same Suite Version Number. The Suite Version Number will be incremented only when significant changes are made to one or more of the components rendering them incompatible with other components from the previous releases of the suite.
- *The Component Major Version Number (Y)* is a version number attributed independently to each component of the suite. It is a subordinate to the Suite Version Number. The Component Major Version Number is reset to zero for each new Suite Version Number and is incremented when the functionality of the component is modified/extended.
- *The Component Minor Version Number (Z)* is a version number that is a subordinate to the Component Major Version Number. The Component Minor Version Number is reset to zero for each new Component Major Version Number and is incremented when small changes are made to the source code that do not reflect a change to the functional requirements of the system. For example, this might include bug fixes or insignificant changes to the functionality of the component.

An update to the software that affects either the Suite Version Number or one of the Component Major Version numbers is defined as a *Major Release*. An update to the software that affects only the Component Minor Version number of either or both components is defined as a *Minor Release*.

2.1 CVS Tagging

In light of the preferred distribution mechanism for the software (that is, NeSCForge), it is appropriate to label versions of QCDgrid components, using the CVS tagging system [1]. Specifically, the following CVS tags will be applied to the two components of QCDgrid:

- **QCDGRID_X_Y_Z** – for the Datagrid and Job Submission component;
- **QCDGRIDMDB_X_Y_Z** – for the Metadata Browser application,

where X, Y, and Z are as defined in the previous section.

Within the NeSCForge repository, three modules are currently defined. These are `src`, `js`, and `java`, containing the Datagrid system, the Job Submission system, and the Metadata Browser, respectively.

Within this module structure, versioning shall be represented by assigning: the `src` and `js` modules with the CVS tag **QCDGRID_X_Y_Z** and the `java` module with the tag **QCDGRIDMDB_X_Y_Z**.

Release versions of the source code for each of these two components (and any others to be defined in the future) will be posted as a TAR archive on the project's NeSCForge account [1] within the table headed "Latest Files Released".

In every QCDgrid component downloaded from NeSCForge, a plain text file named `VERSION` will be found in the root directory of the TAR archive containing the version identifier that corresponds to the particular component.

2.2 New software components

Provision is provided for new software components that may be developed. Any such component will be assigned a new component tag name, following an analogous pattern to that used for the existing components. When a new component attains an appropriate level of quality (the timeframe for which will depend on the component and the project plan), the component will be labelled with the Component Major Version number of X.0.0 and added to the next release of the application suite.

3 Problem resolution

3.1 Reporting a bug or issue

Any problems or issues that arise with either QCDgrid or UKQCD Grid should be registered by generating an appropriate *problem report* (PR) in the NeSCForge Bug Tracker System for the project [2]. Typically, problems relating to the operation or behaviour of the QCDgrid software should be reported as either bugs or functionality requests, while problems/observations pertaining to the operation of the UKQCD Grid should be reported as support requests.

3.2 The lifecycle of a PR

Once a PR is raised with the NeSCForge Bug Tracking System, it follows a standard life-cycle that is defined as follows:

1. **Initial review of the PR:** The NeSCForge system has been configured so that the software development team are notified by e-mail whenever a new PR is raised. Having received such notification of a new PR, the Primary Developer will briefly review the report and determine the urgency and potential impact of the report. Based on this review, the developer will either:
 - a. contact the EPCC Project Leader immediately, for further instruction – this course of action is taken only if the report is determined to be both urgent and high impact.
 - b. prepare a short summary of the PR to be presented and acted upon at the next EPCC project status meeting¹.

This initial review may require further communication with the originator of the PR. To facilitate this process, it is important that the originator of the PR provides some form of contact point – preferably an e-mail address, using the fields provided on the NeSCForge PR form.

2. **Preparation of an action plan for treatment of the PR:** Having completed an initial review of a PR, the EPCC project team will then:
 - a. Categorise the PR as either:
 - ACCEPTED – if the PR is confirmed to be correct and unique (that is not a duplicated of a previously submitted PR);
 - INVALID/DUPLICATE otherwise.
 - b. Assign ACCEPTED PRs to an appropriate member of the project team.
 - c. Agree on a priority and action plan for tackling the PR. The priority is described by a number between 1 (lowest) and 9 (highest) that reflects the urgency with which the project team believe that the report should be resolved. Two forms of action plan may be followed: (i) the assigned team member may be actioned to monitor the PR until a future time at which it is scheduled for resolution in the next software release (this is the typical course of action); (ii) the assigned team member may be actioned to resolve the problem within an agreed timescale (this course of action is usually only followed for high priority problems, in categories 6—9).
3. **Monitoring/resolution of PRs:** At each project status meeting, the list of current open PRs will be reviewed, as follows:
 - New PRs will be assessed, prioritised, and assigned to an individual. An action plan for the PR will be raised and documented in the status meeting minutes.
 - The action plan for each on-going PR will be reviewed and, if necessary, amended to reflect any change in the circumstance of the PR.

¹ EPCC project status meetings typically occur with a weekly frequency.

- Once a PR has been diagnosed, corrected and tested, the originator of the report will be contacted (via e-mail) to confirm that they are now satisfied with the resolution of the problem.
- Once a PR has been resolved to the satisfaction of the project team and the originator, the PR will be closed.

Integral to the resolution/closure of software bugs is the implementation of appropriate testing, where possible. Such tests will be defined in a manner suitable for inclusion in the overall software test suite.

Notes:

1. The NeSCForge Bug Tracking System automatically e-mails the originator of a PR if any change is made to its status.
2. Instructions on how to complete a NeSCForge PR are provided on the NeSCForge web page [2].

4 QCDgrid release schedule

Based on the project roadmap for QCDgrid2 [3], the project team anticipate only infrequent major releases for the QCDgrid software. However, minor releases will be provided that resolve PRs or minor modifications to the software functionality at a four month frequency, typically publishing three new releases during each year. Exceptional minor releases may also be provided from time to time, in response to severe or urgent problems with the QCDgrid software – see Section 4.1 for more details.

One month prior to the each scheduled release, a list of PRs to be tackled will be finalised. Each PR will then be monitored through the EPCC project status meeting, until such time as it is resolved and closed. When all PRs scheduled for resolution prior to the next release have been tackled, a new release of affected components will be published by:

- tagging the affected repository components using the above scheme (see Section 2.1);
- uploading the updated component source code on the NeSCForge website [2].

4.1 Emergency bug fixes

Occasionally PRs will be raised that require urgent attention. Such problems may necessitate an early release in advance of the planned schedule. However, typically the project team will aim to tackle all bugs as part of the normal release cycle.

4.2 Initiation of the new versioning plan

The first release under the new scheme shall occur at the end of April 2005, attributing Version 1.2 tags to the Datagrid/Job Submission system and Metadata Browser: that is, CVS tags **QCDGRID_1_2_0** and **QCDGRIDMDB_1_2_0**, respectively. In advance of this date, a list of bugs to be tackled and resolved shall be finalised by the end of March 2005.

A provisional roadmap for the remainder of QCDgrid project is provided in Figure 1, including some indication of the expected project work plan milestones that will be fulfilled in each release.

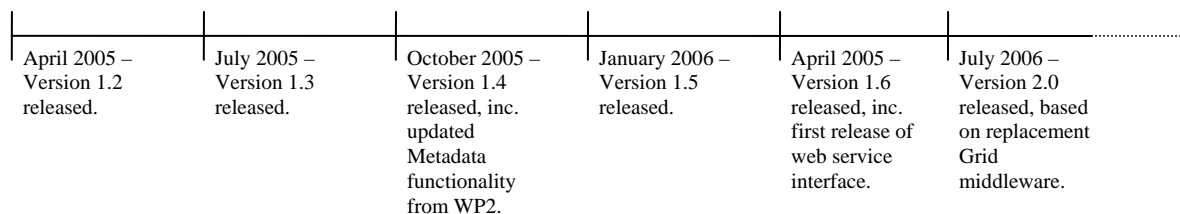


Figure 1: Projected release cycle for the QCDgrid software.

5 Conclusions

This document outlines the processes associated with the release of future updates to the QCDgrid software and the resolutions of problems and issues that may arise with either the software or the UKQCD Grid infrastructure. We define a version naming convention that will allow one to uniquely identify each release of the software. We also establish the procedure by which Problem Reports will be assigned for resolution within the remit of each software release. Finally we present a provisional roadmap for the on-going support of QCDgrid through the duration of the current project work plan.

6 References

- [1] CVS – Concurrent Versioning System, Software application with web homepage at <https://www.cvshome.org/> (February 2005).
- [2] QCDgrid Project Team, *NeSCForge Project Information – Quantum Chromodynamics Grid*, webpage with address <http://forge.nesc.ac.uk/projects/qcdgrid/> (January 2005).
- [3] UKQCD Collaboration, GridPP Project Logbook, available on-line at: http://www.gridpp.ac.uk/qcdgrid/documents/UKQCD_LogBook.pdf (December 2004).