QCDGrid2 Load and Stress Testing

Project Title: QCDGrid2

Document Title: QCDGrid2 Load and Stress Testing

Document Identifier: QCDGRID2-STReq-1.0

Document Filename: LoadStressTestingRequirements.doc

Distribution Classification: Commercial In Confidence

Authorship: Daragh J. Byrne (DJB)

Approval List: QCDgrid Development Team

Distribution List: UKQCD Collaboration

Document History:

Personnel	*Date*	*Summary*	*Version*
DJB	24th November 2004	Version 1.0 for release.	1.0

Contents

1 Introduction 2
1.1 The QCDgrid project 2
1.2 Load and stress testing 2
1.3 Load and stress testing the QCDgrid system 2
2 Requirements 3
2.1 System components and use cases 3
2.2 Gathering requirements 3
2.2.1 Survey responses 4
2.2.2 Analysing survey responses 5
2.2.3 Other issues 6
2.3 Stress and load testing requirements 6
3 Conclusions 7
References 8

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” [1]. Such numerical simulations produce large amounts of data in the form of binary files. The first QCDgrid project developed a software suite running on dedicated hardware (the UKQCD Grid system) that allows UKQCD members to store and share data amongst one another in a straightforward and reliable manner. Metadata in the form of XML files can also be stored on the system. A graphical data browser tool was developed to facilitate ease of use. The software suite also allows users to initiate simulations and post-processing jobs on machines on the grid.

1.2 Load and stress testing

“Load testing generally refers to the practice of modeling the expected usage of a software program by simulating multiple users accessing the program's services concurrently.” [2]. Load testing ensures that a system will operate correctly within performance thresholds at expected loads.

“Stress testing is a form of testing which is used to determine the stability of a given system or entity. It often involves testing something beyond its normal operational capacity in order to observe the results.” [3]. Stress testing often involves subjecting a system to loads that cause it to fail in some manner, or cause its performance to degrade.

The next phase of the QCDgrid2 project aims to develop a test suite that will allow load and stress testing to be carried out on the existing QCDgrid system.

1.3 Load and stress testing the QCDgrid system

The QCDgrid data and compute grid system has been in use since autumn 2002. To date, the load placed on the grid has been low, with only a few of the potential maximum number of users having used the grid. It is anticipated that the load on the system will increase substantially in the future. This increase can be attributed to two main factors:

- The availability of greater compute resources (due to the arrival of the QCDOC machine in late 2004) will enable the production of greater quantities of data;
- The inclusion of the QCDgrid system in the International Lattice Data Grid (ILDG) from 2005.

The QCDOC machine means that a much larger volume of data will be generated than is currently possible. This data will be stored on the grid. The ILDG activities involve the sharing of data stored on the grid with international collaborators. Load and stress testing need to be carried out to ensure that the grid is capable of maintaining these activities. The purpose of this document is to outline the load requirements on the QCDgrid data and compute system.

2 Requirements

2.1 System components and use cases

There are three main components to the software: the data grid, the metadata catalogue, and the job submission system. There are code dependencies, but no operational dependencies – they run as separate components. It is desirable to test each of these components separately.

The QCDgrid system supports the following operations:

- Submit jobs that produce data (e.g. simulations that produce gauge configurations);
- Submit jobs that act on data already present on the grid (e.g. some processing involving previously generated gauge configurations);
- Store data file/files on the grid – these files can be of various (large) sizes;
- Store metadata file/files on the grid;
- Search the metadata catalogue;
- Retrieve data file/files from the grid;
- Retrieve metadata from the grid;
- Delete data files from the grid;
- Delete metadata files from the grid.

In order to adequately estimate the loads placed on the system for testing purposes, it is desirable to know the answers to the following questions:

- How often are jobs that produce configuration data submitted?
- How often are post-processing jobs submitted?
- How often are files written to the grid? What size are these files likely to be?
- How often will the metadata catalogue be searched?
- How often will data files be moved around the grid? Again, what size of transfer?
- How many users will access the system concurrently and what tasks will they be carrying out?

This is a difficult task for the following reasons:

- At present there is a small active user community, which means that it is difficult to predict full load usage patterns;
- The amount of data that will be generated is unclear.

It is generally accepted that the most common operations will involve the reading and writing of data. Job submission facilities have been developed but it is unclear how frequently they will be used.

2.2 Gathering requirements

In order to estimate some of the unknown quantities from above, a user survey was carried out on the UKQCD community. The survey asked users to describe in detail their use of the QCDgrid system; including quantitative data about submission frequency, file size etc. The survey and several reminders

were sent out via the UKQCD-SOFTWARE@jiscmail.ac.uk mailing list. Detailed replies were received from Alistair Hart, Alan Irving, Craig McNeile, Chris Maynard and Jonathon Flynn. Further information and clarification was gathered by email conversation with the respondents.

2.2.1 Survey responses

This section contains summaries of the survey responses returned by five users.

Alistair Hart (University of Edinburgh) submitted information on the current usability of the system. He also outlined two use cases that he sees himself carrying out:

1. Perform a separate analysis on each of a number of gauge configurations. These gauge configurations are selected both by meta-data (beta values, quark masses etc.) and also by the results of previous measurements (e.g. all configurations in that ensemble on which quantity X was measured to have a value in the range 5.9 to 6.1). This could involved tens or hundreds of configurations, which could be fetched just-in-time;
2. Perform an analysis where the results of measurements on a number of configurations are combined together – this may involve files stored on the grid (but not likely).

Alistair's proposed use obviously focuses on the read use case, and may possibly use the job submission use case.

Alan Irving sent a collection of scripts, with directions on how they are used. The first script selects global filenames of files on the data grid. The file names contain physics parameters, and the script does a full list of files on the grid, screening them on some of these physics parameters using *grep*. The end result is a text file containing the name of a file of interest on each line.

The next step in the analysis chain is to grab individual configurations from the grid using the QCDgrid "get-file-from-grid" command in a loop. N configurations are grabbed at a time. N is typically 8-12 but more usually 2-3 – although fundamentally files are pulled across one at a time. The configurations are placed in a job queue and each is processed on a single node of a local cluster. Processing typically takes 1hr 20mins. The queue is topped up from time to time (another N are grabbed) and put in the queue. This continues until all configurations of interest are processed. The configuration files are typically 125 Mb in size. Typically, Alan will pull 8-12 in one go.

Craig McNeile operates in a similar manner. Files are pulled from the grid in batches as above and processed. Currently configuration sizes are 75Mb. Craig estimates file sizes of 500Mb for the first QCDOC runs. Files are generally pulled down in batches of up to 20.

Chris Maynard estimates that a write rate of 1Tb per day is not unreasonable. This would be a maximum production figure. Chris is unsure whether this is all in one go, several large files or many smaller files. Scope should be left to test all possibilities. Chris also made the point that in some ways the usage of the system will be constrained by how easy it is to move data around – if it's difficult the community will have to consider their actions more. Chris provided a rough estimate that there would be about 30 active users of the system, and said that a concurrent figure of 10 was not unreasonable, although this was a very rough estimate. Craig McNeile stated that a figure of 10Gb per day would perhaps be more appropriate.

Jonathan Flynn responded but was unable to present usage patterns as they have not been using the grid. He will give more input later.

2.2.2 Analysing survey responses

An analysis of the responses to the email survey reveals the following:

- The “read data” and “write data” use cases are the most popular and have been given the most consideration by the user community;
- Little consideration had been given to the use of the metadata catalogue and the job submission system, though at least some users intended to carry out post-processing jobs on local nodes;
- Data will be written to the grid at the (approximate) maximum rate of 1 Tb per day, in files ranging in size from 75 Mb to 500 Mb, and possibly bigger should bigger lattices be used; although this rate might be closer to 10 Gb per day.
- No consideration has been given as to the impact the ILDG activities will have on the performance of the grid;
- Configuration file sizes will vary from about 75Mb to 500Mb, with larger files being entirely possible.

The answers to the questions posed in section 2.1 are thus:

How often are jobs that produce configuration data submitted?

This is unclear and cannot be answered until QCDOC is in production. However, data may be produced at a rate of up to 1Tb a day. In any event, this may not occur via the QCDgrid job submission software.

How often are post-processing jobs submitted?

This again is unclear and will not be answerable until the grid is in wider use.

How often are files written to the grid? What size are these files likely to be?

As stated a write rate of 1Tb per day has been quoted. The configuration files are likely to be anything from 75Mb to 500Mb, with the possibility of larger files.

How often will the metadata catalogue be searched?

This is very unclear, with the system so far having been used lightly and mainly for demonstration purposes.

How often will data files be moved around the grid? Again, what size of transfer?

Again, there is no clear answer to this question. Files will be written to the grid once only, after they have been produced. They may then be moved around the grid any number of times following the “read data” use case.

How many users will access the system concurrently and what tasks will they be carrying out?

A guideline figure of ten to twenty users may be using the grid at any one time. Much of the activity will follow the “read data” and “write data” use cases.

As can be discerned from the feedback, exact load requirements are difficult to pin down for the grid. We thus adopt the position that testing will be carried out to investigate where the load limits of the grid lie, rather than to see whether it meets a particular set of (unclear) requirements. In the cases that we do have figures to work with, we will test against the “worst case” loads provided by the figures, and above those loads to see where the limits lie.

The following load/stress testing strategy is therefore proposed:

- The datagrid write characteristics can be tested under the loads specified in the results of the user survey, and above those loads (i.e. at and above the 10 Gb/day and 1Tb/day limits quoted, or to breakage);
- The datagrid read use case can be tested to breaking point, i.e. to investigate what the maximum loads are. At the very least, the "read data" use case should support one user pulling down 12 x 500 Mb configurations every hour, as per the feedback by Craig McNeile and Alan Irving;
- The job submission system and metadata catalogue can be tested to see what there maximum loads are, since the anticipated loads are not known.

2.2.3 Other issues

The system has other degrees of freedom that have not been considered in the above analysis.

The system is a multi-user system. There are approximately sixty people involved in the UKQCD consortium. Chris Maynard gave a very rough estimate that no more than half of these people will become heavy users of the UKQCD Grid, probably 10-20. The test suite needs to simulate concurrent use of the system by multiple users, using different combinations of parameters such as file sizes.

At present, the impact of the ILDG on the system has not been assessed. It is difficult to speculate on the best way to incorporate load testing for the ILDG without conducting a wide scale survey of the ILDG members. Also, the architecture for the UKQCD implementation of the ILDG web services has not yet been defined, meaning it is unclear as to how best to simulate it. It is possible to say that the ILDG load will be mainly a result of reading from the datagrid. We propose that the stress test framework developed for this phase of the project be flexible enough to use when these issues have been resolved. We will not carry out specific testing for the ILDG activities until such a time as these issues are resolved. However, the results of the stress tests will be able to inform future discussions about the amount of load the ILDG systems will be able to place on the grid without causing problems.

A further issue that has not been considered is the time it takes for a piece of data to become available for retrieval once it is placed on the grid. This is a quantity of interest and should be measured.

2.3 Stress and load testing requirements

From the above survey and analysis, we can define a requirement for a suite of software that will load test the QCDgrid system components. This software will most likely take the form of a set of shell scripts (the exact form of the software will be described in a separate document). The test suite has the following requirements:

R-1 A test grid needs to be created in order to run the load tests without disrupting the running of the production grid. This has already been put in place by EPCC.

R-2 The test suite should allow the "write data" use case to be run in a flexible manner. The suite should allow the simulation of writing data to the grid in a variety of file sizes, by a variable number of users and over sustained periods of time, such as one whole day, one hour, ten minutes etc. The test suite will exercise the 10 Gb per day and 1 Tb/day estimate for data

production in a number of ways, using different combinations of file sizes. The test will also be run above this rate.

R-3 The test suite should allow the “read data” use case to be run in a flexible manner. The suite should again be flexible with respect to file size, number of users, period of time, number of reads in a period of time etc. The test should also examine performance metrics, such as the time it takes to read files of certain sizes. The test will be run to pull across at least 12 configurations of 500Mb size, and at higher loads until the system fails.

R-4 The test suite should exercise the job submission system. The suite should allow the automated submission of jobs taking a specified amount of time at a given rate, by a certain number of users.

R-5 The test suite should allow the server part of the metadata catalogue to be placed under various loads. This part of the suite should be flexible with respect to number of queries, expected size of data returned by query, rate of query and number of concurrent users. This part of the suite will leverage existing client side code.

R-6 A data generation tool for creating files of various sizes needs to be written or located.

R-7 The test system should examine how long it takes a piece of data to become available on the grid after being submitted.

R-8 A full report on the load and stress testing carried out needs should be written after testing.

3 Conclusions

Load and stress testing a system with unknown loads is difficult. We have adopted a policy of testing to and above limits where they are known, and testing to failure where limits are not known.

A suite of software implementing the above requirements will be designed and developed. The suite will consist of scripts that call the existing QCDgrid client software. The results of the stress testing will be distributed to the community.

References

[1] UKQCD Collaboration home page http://www.ph.ed.ac.uk/ukqcd/
[2] Wikipedia definition of load testing http://en.wikipedia.org/wiki/Load_testing
[3] Wikipedia definition of stress testing http://en.wikipedia.org/wiki/Stress_testing