

PGPGrid Rendering Environment Survey

Project Title: PGPGrid Project

Document Title: PGPGrid Rendering Environment Survey

Document Identifier: PGPGrid-RenderingEnvironment-Survey-v1.0

Distribution Classification: Public Document

Author: Ali Anjomshoaa

Approval List:

EPCC: Kostas Kavoussanakis, Gavin Pringle

Distribution List:

EPCC: Kostas Kavoussanakis, Gavin Pringle
 PGP: Carl Goodman, Peter Stansfield
 3D-Matic: Paul Cockshott

Document History:

Personnel	Date	Summary	Version
AA	17/07/2003	First Draft	0.1
AA	28/07/2003	Second Draft – More information added to Sections 5 and 6.	0.2
AA	30/07/2003	Third Draft – Have added Section 3, Section 7, and References for release to EPCC for comments.	0.3
AA	31/07/2003	Forth Draft – Have addressed comments from Gavin Pringle.	0.4
AA	29/08/2003	Fifth Draft – Have addressed comments from Kostas Kavoussanakis and results of discussion with Carl Goodman, and added Acknowledgements.	0.5
AA	08/09/2003	Sixth Draft – Have added more background information to the project in Section 2. Have also added Section 5.1.7.	0.6
AA	15/09/2003	Seventh Draft – Have addressed comments from Gavin Pringle. Have added Section 4 following feedback from Carl Goodman.	0.7
AA	21/11/2003	Eighth Draft – Have addressed comments from Gavin Pringle and Kostas Kavoussanakis. Have updated Section 4.3.1 following further feedback from Carl Goodman. For release to project partners.	1.0

Contents

1	Introduction	3
2	The PGPGrid Project.....	3
3	The Rendering Environment Survey	3
4	PGP's Rendering Mode of Operation	3
4.1	Model Creation.....	3
4.2	Addition of Textures, Lighting, and Detailed Information	3
4.3	Submission of Rendering Jobs to the PC Render Farm.....	3
4.3.1	The Render Manager	3
4.3.2	The Queue Manager	3
4.4	Interactive Quality Assurance	3
4.5	Key 3D Studio Max Features.....	3
4.6	Other Key PGP Practices	3
5	Rendering Computing Resources	3
5.1	Available Computing Resources.....	3
5.1.1	The Ness SMP Cluster.....	3
5.1.2	The HPCx Supercomputer	3
5.1.3	The HPCx Development Nodes.....	3
5.1.4	The Blue Dwarf Power4 IBM Node.....	3
5.1.5	The ScotGRID Linux Cluster	3
5.1.6	The University of Edinburgh's Informatics Beowulf Cluster.....	3
5.1.7	The EPCC Microsoft Windows PC Cluster.....	3
5.2	Discussion of the Rendering Computing Resources	3
6	Rendering Software	3
6.1	Discussion of the Rendering Software.....	3
7	Recommendation for Overall Rendering Environment	3
7.1	Recommendations for Further Investigation	3
	Appendix: POV-Ray to 3D Studio Max Translators.....	3
	Acknowledgements	3
	References	3

1 Introduction

This report is a public document, which describes a survey of hardware and software to enable a Virtual Organisation (VO) to apply Grid technologies to the production of computer-generated animation in the rendering process.

This report begins by describing the Pepper's Ghost Productions Grid (PGPGrid) project VO. It then goes on to describe the results of the survey, which was carried out between July and September 2003.

This report is one of the deliverables of "Work Package 6" of the PGPGrid project, which is described in the PGPGrid project proposal [1].

2 The PGPGrid Project

The PGPGrid project is a collaboration between:

- Pepper's Ghost Productions Ltd. (PGP);
- 3D-Matic Laboratory of The University of Glasgow (3D-Matic); and
- EPCC of The University of Edinburgh (EPCC).

The PGPGrid project aims to bring together a VO involving these partners. The goal of this VO is to apply Grid technologies to the production of computer-generated animation, where a unique method of motion-capture is applied to the animation process. This endeavour will enable the VO to study the feasibility of the application of this approach to computer-generated animation and the possible impact of Grid technologies on this industry.

The 3D-Matic laboratory produces computer-generated three-dimensional (3D) mesh models of the motion of real human subjects. Motion-capture is achieved using eight sets of three TV cameras, two monochrome and one colour, which are positioned statically around the subject in a studio. The produced mesh models can be rotated in any direction in three dimensions. This has the effect of producing just about any camera angle. This is equivalent to an animator being able to position a camera anywhere they wish, after the process of motion-capture, resulting in enormous creative opportunities while cutting the motion-capture overheads of re-shoots.

The production of these 3D mesh models is a two step process using in-house software applications. These processes are called *ranging* and *modelling*. Figure 1 shows the overall process for the production of a model, for one frame of motion-capture, using the ranging and modelling processes, in that order. The computer-generated mesh models that are produced by 3D-Matic can then be used by PGP to create animation by the addition of scenery and textures to the models, prior to rendering each frame using third party rendering software.

The processes of ranging and modelling are both computationally intensive tasks. For each captured frame, eight processes are used for ranging, one for each pair of monochrome cameras from the eight sets that are used for motion-capture. During the ranging process, the data from each pair of monochrome cameras is analysed in order to calculate the positions of a set of points on a grid, which was projected onto the subject during motion-capture, and as viewed by the pair of cameras. A method of parallax correction is used for this calculation. The ranging data from the eight ranging processes are then merged in a single modelling process, to produce three dimensional mesh models for each single frame. It should be noted that, the colour camera in each of the eight sets of cameras is used for the capture of skin textures and tones of a subject, which can subsequently be used in the rendering process. These processes can all be performed

remotely, with respect to 3D-Matic, on distributed resources that may be accessed using Grid middleware such as the Globus Toolkit.

In addition, the process of rendering, which is also significantly computationally intensive, can be performed remotely, with respect to PGP, on distributed resources that may be accessed using Grid middleware. Rendering can be performed on separate frames concurrently. Depending on the rendering software used, rendering may also be performed on segments of a frame in parallel.

The parallel natures of both the ranging and modelling processes, and that of rendering, added to the fact that frames of motion-capture and animation can be produced in parallel, result in an animation production process that can take advantage of a rich environment of distributed resources for these processes to run in parallel. The ability to undertake some of these parallel processes in an environment of heterogeneous platforms of computing resources, using state of the art Grid middleware, will be of particular interest.

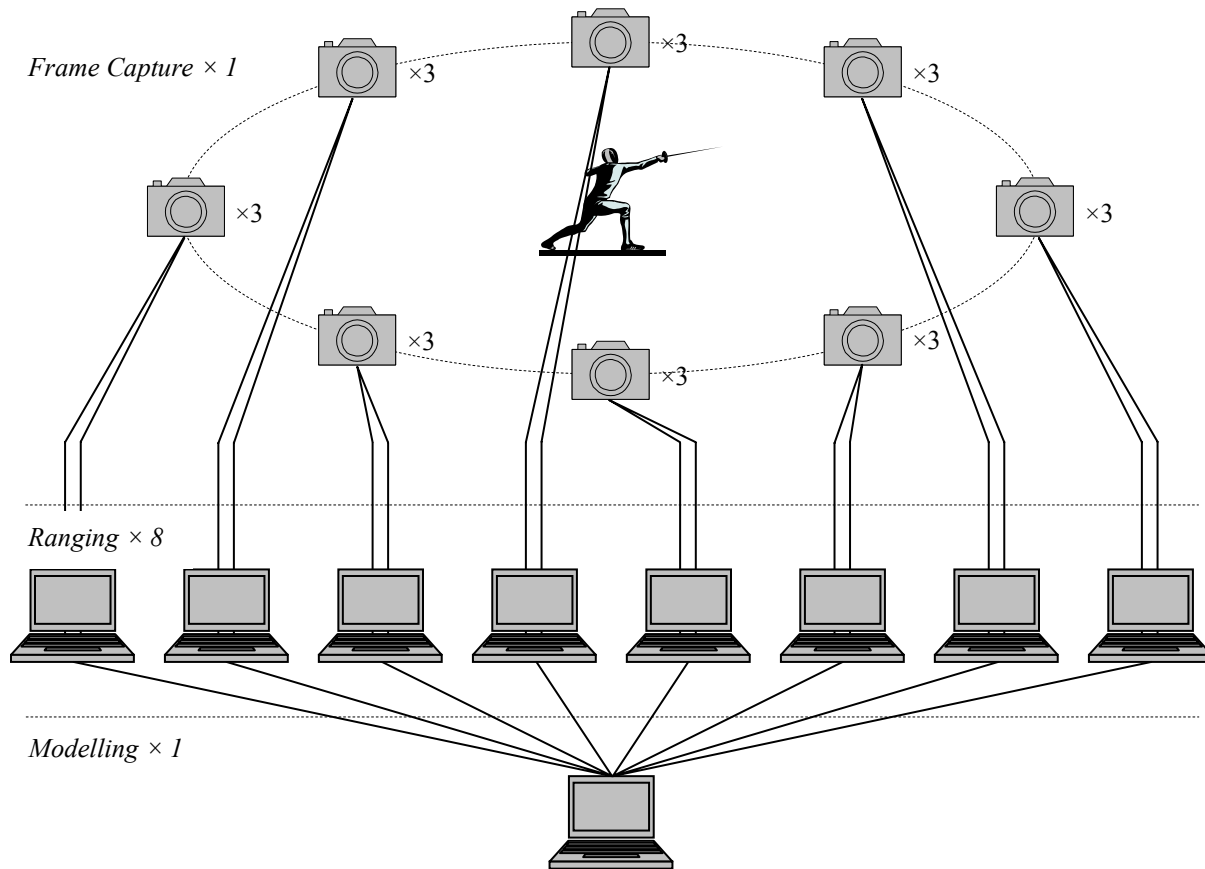


Figure 1: The schematic of the system used by 3D-Matic for motion-capture. At the top, eight sets of three cameras are shown, two monochrome and one colour. Also shown is the current set up for the process of *ranging*, where data from the two monochrome cameras in each set are analysed as a separate process for each frame. In the process of *modelling*, the ranging data from all of the eight ranging processes performed per frame, are merged into a single model for that frame.

The aim of this project is to determine the following:

- whether or not the formation of VOs, such as the PGPGrid VO, are a feasible method for future computer-generated animation production;
- whether the unique motion-capture technology offered by 3D-Matic can be used within such a VO in the animation process; and
- whether it is possible to undertake the processes of ranging and modelling, as required by 3D-Matic for their model production, and the process of rendering, as required by PGP, within a Grid environment using middleware technologies such as the Globus Toolkit.

With respect to the aim of utilising a Grid environment, the goal will be to determine whether this will enable the use of a distributed, and perhaps heterogeneous, environment of computing resources. In addition, whether these processes, run on such a Grid, may be controlled remotely. And finally, to identify the particular customisations that are required in order to take advantage of a Grid environment.

Further background on the project partners, details of their roles in the VO, and further details of the PGPGrid project, are contained in the PGPGrid project proposal [1].

3 The Rendering Environment Survey

The remainder of this report examines the choices of rendering environments for the PGPGrid project. The “rendering environment”, in this case, includes the rendering software and the associated resources required for that software. The required resources include the computational platforms and their associated hardware and software. Associated hardware includes such things as disk storage and networking hardware, and the associated software includes the operating system (OS) of the resource and any compilers and libraries that may be required by the rendering software.

The PGPGrid project aims to investigate the feasibility of performing the process of rendering over a distributed and heterogeneous Grid of computing resources. The aim is to provide access to these resources using Grid middleware that will allow for a scalable solution upon demand. In addition, it is important to show that access to these resources can be provided without direct access to them and that rendering processes can be initiated and controlled remotely.

4 PGP's Rendering Mode of Operation

This section describes the current mode of operation used by PGP for rendering using the 3D Studio Max application software. This mode of operation is described in a number of steps in the following subsections.

4.1 Model Creation

Computer-generated models are created by animators on their desktop machines prior to rendering. This is generally done using the 3D Studio Max application software. Other variations of this process include the digitisation of sculptures using laser scanning devices, or point-digitising systems. Although not common practice, systems also exist for the scanning of live performers. The 3D-Matic installation provides a new method for doing this.

Once the basic computer-generated model has been created, the data describing the model are saved to disk on a dedicated server, in the 3D Studio Max data format. A simple file system is used for data file management.

4.2 Addition of Textures, Lighting, and Detailed Information

After the creation of computer-generated models and the storage of their data files to disk, animators work on creating further details for the models. This work is undertaken on the animator's desktop machine, using, for instance, Adobe's Photoshop software.

This detail creation involves the creation and selection of textures, lighting, skins and hair, and character animation and surface deformation data. Textures are designed to affect certain specific characteristics of materials, which will tend to include ambient, diffuse and specular colour channels, specular levels, opacity, bump, displacement and anisotropy.

Once this detail has been created and selected, the information is added to the basic model using 3D Studio Max and again saved to the file system. Version control is carried out manually at this point.

Models and texture data files are stored independently from each other. Model data files, once rigged for vertex deformation, are generally in the range of 5 to 50 megabytes (MB) in size. The associated texture data files for a complex scene, are generally held in Truevision Targa format and can often range from hundreds of megabytes into the gigabyte range, unless procedural shaders are being employed.

4.3 Submission of Rendering Jobs to the PC Render Farm

Once the models and their details have been created and saved in data files, rendering jobs may be submitted. These jobs are submitted to the Microsoft Windows PC render farm that is managed using the 3D Studio Max Remote Rendering software.

To submit a rendering job, the identity of the 3D Studio Max Render Manager, including the IP address of the server machine on which it runs, must be specified. The Render Manager, along with the 3D Studio Max Queue Manager, must be allocated servers to run on. These servers can either be standalone machines, or one of the farm nodes. It is recommended, however, that the Render Manager and Queue Manager servers do not also serve as worker nodes on the farm.

4.3.1 The Render Manager

The charge of the Render Manager is that of distributing rendering jobs to farm nodes and allocating animation frames for them to render. A rendering job consists of the entire set of frames that make up the animation. The rendering job is submitted to every node in the rendering farm, before each node is allocated the frame that it will render. Once a node has finished rendering its allocated frame, it will be allocated another frame by the Render Manager. For the rendering of each frame, each node within the rendering farm pulls, or uploads, 3D Studio Max files to itself in order to undertake the rendering of that frame.

In addition, each farm node requires the texture-map data file for the entire animation job, for the rendering of each frame. Although the rendering job may be configured so that the texture-map file is sent to the nodes when the job is submitted to them, this adds undue complexity. A pull model, whereby a rendering node picks up the texture-map file once the rendering job has been submitted to it, is the alternative option. This pull model, however, results in a network bottleneck due to contention between the nodes accessing the texture-map file. This is caused by limited network bandwidth. The pull model that is currently used,

therefore, uses a job configuration, where an animator submits the rendering job and allocates frames in batches of four to eight to the same number of farm nodes. Once those farm nodes have fetched the texture-map file and started the rendering of the frames that they have been allocated, the next batch of four to eight rendering jobs can be submitted, and frames be allocated by the animator, to the next set of four to eight nodes. This configuration avoids the network bottlenecks that would be caused by the fetching of the texture-map file by the first and the second set of nodes, and so on. Once rendering of frames has started on every farm node, they each have the texture-map data file for the rendering job. The animator is then no longer required to submit jobs and allocate frames in batches, and the Render Manager takes sole charge of allocating further frames to farm nodes as they become available.

4.3.2 The Queue Manager

The Queue Manager is an interface that monitors the activity of a specific set of Rendering Managers, which are identified by their IP addresses, and which then tracks the reported performance of the individual rendering servers assigned to each manager. Using the Queue Manager interface, jobs can be re-assigned, have their priorities changed, signal completion, or be deleted.

4.4 Interactive Quality Assurance

Animators are able to undertake interactive quality assurance (QA) by using the 3D Studio Max Frame Buffer, which shows dynamic updates of the frame being rendered. This is used to find how well or how far a job has progressed. The animator monitoring progress has the ability to abort the rendering job if the result is not of acceptable quality. Model data can then be reconfigured and the job resubmitted.

This level of interactivity is not absolutely required, since it is not possible to provide this level of QA round-the-clock. It is sufficient to be able to scrutinise finished frames after rendering of those frames has completed, and to allow the animation rendering job as whole to continue or to abort it accordingly.

4.5 Key 3D Studio Max Features

The 3D Studio Max application software does not distribute single frames for rendering. It distributes animation tasks across the rendering farm as a set of frames at one time. The Render Manager distributes the whole sequence to each node of the render farm, each of which in turn would fetch an appropriate frame for rendering when it is ready. This is explained in more detail in Section 4.3.1.

Note, that the Queue Manager would be initiated by an animator into stopping a job, if a rendered frame was found to be unsatisfactory.

4.6 Other Key PGP Practices

At PGP, rendering may be undertaken 24 hours a day. Lost time due to problems occurring at the rendering phase of production, is estimated and accounted for at the project planning stages. The overhead budget for failures occurring during the rendering phase, is approximately 20%.

5 Rendering Computing Resources

In order to determine the rendering environment for the PGPGrid project, it is necessary to first determine a suitable computing platform. This task involves the study of the specifications and particulars of the available computing resources.

5.1 Available Computing Resources

The computing resources that have been identified are:

- The University of Edinburgh's Sun SMP HPC resource Ness;
- The HPCx resource based at Daresbury Laboratory;
- The HPCx development nodes based at Daresbury Laboratory;
- The National e-Science Centre's (NeSC) Blue Dwarf Power4 IBM node;
- The ScotGRID Linux cluster at Glasgow University;
- The University of Edinburgh's Informatics Beowulf Linux cluster; and
- The EPCC Microsoft Windows PC cluster.

The following sections include the particulars of each of these possible resources.

5.1.1 The Ness SMP Cluster

Details of the University of Edinburgh's Ness SMP cluster from Sun, can be found in the user guide for this system, which is located at:

<http://www.epcc.ed.ac.uk/computing/services/sun/documents/hpc-intro/html/index.html> .

It has been confirmed that this resource can be made available to the project. Further, it may be possible to get exclusive access to the machine at specific times, if required.

5.1.1.1 Ness Specifications and Configuration

Ness SMP Cluster	
Hardware	
System Vendor	Sun Microsystems
Machine Type/Model	Sun Fire 15K Server
Number of CPU Nodes	4 interactive + 48 batch
CPU Architecture and Speed	UltraSPARC III, 900 MHz
RAM Memory Type and Size	4 GB interactive + 48 GB batch
Cache Levels and Size	L1: 64KB, L2: 8MB
Disk Available	3 × 24 GB disks currently (6 th August 2003) available as workspace.
Switching Technology	150 MHz Sun Fireplane, redundant 18×18 crossbar interconnect.
Software	
Operating System and Version	Solaris 2.9
Compilers and Versions	Supported: Fortran, C, C++, Java, MPI, OpenMP, HPF Compilers: f90; Forte Developer Fortran 95 7.0 cc; Forte Developer C 5.4 CC; Forte Developer C++ 5.4 javac/java; 1.4.0_00(-b05) mpf90/mprun; Forte Developer Fortran 95 7.0 mpcc/mprun; Forte Developer C 5.4 pghpf; Portland Group, Inc., Release 2.4-4
Batch System and Version	Sun GridEngine v5.3p2
Exclusive Priority Availability	Negotiable
Maximum Job Time (wall clock)	By default: 1 – 8 processors: 48 hrs, 9 – 32 processors: 12 hrs
Maximum Number of CPUs/Job	By default: 32
Minimum Number of CPUs/Job	By default: 1

5.1.1.2 Ness Access Details and Support

Ness SMP Cluster	
System Owner	University of Edinburgh
System Administrator	EPCC
System Availability Policies	Negotiable (at cost to EPCC)
System Maintenance Frequency	Irregular maintenance
System Upgrade Schedule	Non planned
System Reliability (% uptime)	No statistics available at the time of publication.
Allocation Unit (AU) Cost	Cost has not been explicitly set. (1AU = 0.687 CPU Hours)
CPU Hour Cost	Cost has not been explicitly set.
Available CPU Hours per Month	The system produces an average of 50,000 AUs per month.

5.1.2 The HPCx Supercomputer

Details of the HPCx machine can be found from the HPCx website at:

<http://www.hpcx.ac.uk/services/> .

Access to this machine is not possible without explicit purchase of CPU time.

5.1.2.1 HPCx Specifications and Configuration

HPCx Supercomputer	
Hardware	
System Vendor	IBM
Machine Type/Model	POWER4 System based pSeries 690 Regatta nodes
Number of CPU Nodes	1280 (40 Regatta nodes × 4 SMP nodes ("LPARs") × 8 POWER4 processors)
CPU Architecture and Speed	POWER4, 1.3GHz
RAM Memory Type and Size	8 GB (shared by 8 CPUs) on each "LPAR"
Cache Levels and Size	L1: 32KB data, 64KB instruction L2: 1.44MB (shared by 2 CPUs) L3: 32MB (shared by 32 CPUs)
Disk Available	18 TB of total disk on the system.
Networking Infrastructure	1 gigabit-per-second (Gbps), although this is currently limited to 0.4 Gbps due to fire wall constraints. (This will be upgraded to a 10 Gbps network in the first half of 2004)
Switching Technology	IBM Colony (dual plane, omega network, packet based interconnect), between 8 SMP nodes, 350MB/s peak unidirectional bandwidth.
Software	
Operating System and Version	AIX 5L
Compilers and Versions	Supported: Fortran, C, C++, Java, MPI, OpenMP Compilers: xlf90; XL Fortran for AIX v08.01.0001.0000 xlc; C for AIX v6.0.0.2 xlC; C for AIX v6.0.0.2 javac/java; 1.3.0 mpxlf90/poe; XL Fortran for AIX v08.01.0001.0000 mpcc/poe; C for AIX v6.0.0.2 mpCC/poe; C for AIX v6.0.0.2
Batch System and Version	LoadLeveler for AIX 5L
Exclusive Priority Availability	None
Maximum Job Time (wall clock)	By default: 1 – 8 processors: 6 hrs, By default: 128 – 1024 processors: 12 hrs
Maximum Number of CPUs/Job	By default: 1024
Minimum Number of CPUs/Job	By default: 1 (must pay in blocks of 8 CPUs, i.e. 1 "LPAR" SMP node)

5.1.2.2 HPCx Access Details and Support

HPCx Supercomputer	
System Owner	UoE HPCx Ltd.
System Administrator	University of Edinburgh/Daresbury Laboratory
System Availability Policies	CPU time for sale
System Maintenance Frequency	Once a fortnight
System Upgrade Schedule	Major system upgrade planned for July 2004.
System Reliability (% uptime)	More than 99%
Allocation Unit (AU) Cost	<ul style="list-style-type: none"> ○ £0.38759 (until Jul. 2004), (1AU = 0.395 CPU Hours) ○ £0.20936 (estimated for Jul. 2004 – Nov. 2006), (AU:CPU Hour ratio not available)
CPU Hour Cost	<ul style="list-style-type: none"> ○ £0.98525 (until Jul. 2004) ○ Not available (Jul. 2004 – Nov. 2006)
Available CPU Hours per Month	Based on purchase of CPU time.

5.1.3 The HPCx Development Nodes

It has been confirmed that this resource can be made available to the project, though the amount of dedicated time will be decided upon receipt of a formal application.

It must be noted that this machine is “at risk”, since it is a testing platform for applications that are to be deployed on the HPCx machine described in Section 5.1.2, and can be brought down without reasonable warning.

5.1.3.1 HPCx Development Nodes Specifications and Configuration

HPCx Development Nodes	
Hardware	
System Vendor	IBM
Machine Type/Model	POWER4 System based pSeries 690 Regatta nodes
Number of CPU Nodes	64 (2 Regatta nodes × 4 SMP nodes ("LPARs") × 8 POWER4 processors)
CPU Architecture and Speed	POWER4, 1.3GHz
RAM Memory Type and Size	8 GB (shared by 8 CPUs) on each "LPAR"
Cache Levels and Size	L1: 32KB data, 64KB instruction L2: 1.44MB (shared by 2 CPUs) L3: 32MB (shared by 32 CPUs)
Disk Available	This information was unavailable at the time of publication.
Networking Infrastructure	1 gigabit per second (Gbps), although this is currently limited to 0.4 Gbps due to fire wall constraints. (This will be upgraded to a 10 Gbps network in the first half of 2004)
Switching Technology	IBM Colony (dual plane, omega network, packet based interconnect), between 8 SMP nodes, 350MB/s peak unidirectional bandwidth.
Software	
Operating System and Version	AIX 5L
Compilers and Versions	Supported: Fortran, C, C++, Java, MPI, OpenMP Compilers: xlf90; XL Fortran for AIX v08.01.0001.0000 xlc; C for AIX v6.0.0.2 xlC; C for AIX v6.0.0.2 javac/java; 1.3.0 mpxlf90/poe; XL Fortran for AIX v08.01.0001.0000 mpcc/poe; C for AIX v6.0.0.2 mpCC/poe; C for AIX v6.0.0.2
Batch System and Version	LoadLeveler for AIX 5L
Exclusive Priority Availability	Dependent on application
Maximum Job Time (wall clock)	Not Applicable
Maximum Number of CPUs/Job	64
Minimum Number of CPUs/Job	1

5.1.3.2 HPCx Development Nodes Access Details and Support

HPCx Development Nodes	
System Owner	UoE HPCx Ltd.
System Administrator	University of Edinburgh /Daresbury Laboratory
System Availability Policies	EPCC CPU time available.
System Maintenance Frequency	Uncertain, believed to be frequent since this is a development machine.
System Upgrade Schedule	Not Available
System Reliability (% uptime)	Unknown as records are not kept.
Allocation Unit (AU) Cost	Not Applicable
CPU Hour Cost	Not Applicable
Available CPU Hours per Month	Not Applicable

5.1.4 The Blue Dwarf Power4 IBM Node

This machine is administered by the system administration team at the National e-Science Centre (NeSC) in Edinburgh. Indications are that this machine can be made available for this project and that there is a large amount of disk space available for the large data required by PGPGrid.

5.1.4.1 Blue Dwarf Specifications and Configuration

Blue Dwarf Power4 IBM Node	
Hardware	
System Vendor	IBM
Machine Type/Model	POWER4 System based pSeries 690 Regatta node
Number of CPU Nodes	14 POWER4 processors
CPU Architecture and Speed	POWER4, 1.3GHz
RAM Memory Type and Size	32 GB (shared by 14 CPUs)
Cache Levels and Size	Information not available from NeSC.
Disk Available	880 GB (can accommodate up to 200 GB for PGPGrid)
Networking Infrastructure	Information not available from NeSC.
Switching Technology	Information not available from NeSC.
Software	
Operating System and Version	AIX 5.1
Compilers and Versions	Information not available from NeSC.
Batch System and Version	Not Applicable
Exclusive Priority Availability	None
Maximum Job Time (wall clock)	Not Applicable
Maximum Number of CPUs/Job	14
Minimum Number of CPUs/Job	1

5.1.4.2 Blue Dwarf Access Details and Support

Blue Dwarf Power4 IBM Node	
System Owner	NeSC
System Administrator	NeSC
System Availability Policies	Available on demand – may change at a later date depending on level of demand from other users.
System Maintenance Frequency	Irregular maintenance as required, although the following have been stated as upcoming: <ul style="list-style-type: none"> • one day to move the machine, • OS upgrade from AIX 5.1 to AIX 5.2. (dates for above are unknown)
System Upgrade Schedule	This is not currently known by NeSC.
System Reliability (% uptime)	“very stable and fault tolerant” – quoted by NeSC system admin.
Allocation Unit (AU) Cost	Not Applicable
CPU Hour Cost	Not Applicable
Available CPU Hours per Month	Not Applicable

5.1.5 The ScotGRID Linux Cluster

This machine is administered by the ScotGRID collaboration and is based at the University of Glasgow. Indications are that this machine can be made available for this project.

Utilisation statistics for this machine are available from the ScotGRID website at:
<http://www.scotgrid.ac.uk> .

The accumulated statistics indicate that the machine has been, and is, well utilised by a number of particle physics projects.

5.1.5.1 ScotGRID Linux Cluster Specifications and Configuration

ScotGRID Linux Cluster	
Hardware	
System Vendor	IBM
Machine Type/Model	Pentium III based xSeries 330 nodes
Number of CPU Nodes	59 nodes
CPU Architecture and Speed	Dual 1.0GHz PIII per node
RAM Memory Type and Size	2GB per node
Cache Levels and Size	256KB Level-2 cache
Disk Available	Up to 5TB long term, the timescale for an upgrade is unknown.
Networking Infrastructure	The storage, head (user) and management nodes operate on a 100baseT private network to three Cisco switches. These three Cisco switches trunk to an eight port Gigabit only switch with connections to file servers and the Internet. The only access to the Internet is via IPChains IP Masquerading on the master node that manages the cluster.
Switching Technology	Currently, maximum Internet speed for any one box is only 100 Mb per second. Three 3548XL and one 3508XL Cisco switches.
Software	
Operating System and Version	Red Hat Linux 7.2
Compilers and Versions	Standard set available with Red Hat Linux 7.2, including the GNU Compiler Collection.
Batch System and Version	Maui
Exclusive Priority Availability	None
Maximum Job Time (wall clock)	No current upper limit
Maximum Number of CPUs/Job	1 (Parallel (MPI) jobs are not supported)
Minimum Number of CPUs/Job	1

5.1.5.2 ScotGRID Linux Cluster Access Details and Support

ScotGRID Linux Cluster	
System Owner	ScotGRID
System Administrator	ScotGRID, University of Glasgow
System Availability Policies	Available on demand
System Maintenance Frequency	Irregular maintenance as required.
System Upgrade Schedule	Additional resources are planned for installation at the start of 2004. Significant downtime is expected when the cluster has to be rebuilt and the new equipment integrated.
System Reliability (% uptime)	Approximately 95%
Allocation Unit (AU) Cost	Not Applicable (no charging model used)
CPU Hour Cost	Not Applicable (no charging model used)
Available CPU Hours per Month	Not Applicable (no charging model used)

5.1.6 The University of Edinburgh's Informatics Beowulf Cluster

This cluster consists of a 64 node GX240 DELL sub-cluster and a 16 node Workstation530 DELL sub-cluster. Each sub-cluster contains a head node. Although at present it is possible to access each node individually, it has been expressed that there is a need to provide a batch system, possibly PBS, on each head node. The batch system would have to use the Kerberos authentication mechanism to submit jobs to the cluster nodes.

5.1.6.1 Informatics Beowulf Cluster Specifications and Configuration

University of Edinburgh Informatics Beowulf Cluster	
Hardware	
System Vendor	DELL
Machine Type/Model	<ul style="list-style-type: none"> ○ DELL GX240 × 64 nodes ○ DELL Workstation530 × 16 nodes
Number of CPU Nodes	<ul style="list-style-type: none"> ○ GX240 cluster: 1 head node + 63 nodes ○ Workstation530 cluster: 1 head node + 15 nodes
CPU Architecture and Speed	<ul style="list-style-type: none"> ○ GX240 cluster: 1.8 GHz P4 ○ Workstation530 cluster: 1.7 GHz Xeon
RAM Memory Type and Size	<ul style="list-style-type: none"> ○ GX240 cluster: 1 GB per node ○ Workstation530 cluster: 2 GB per node
Cache Levels and Size	Information not provided.
Disk Available	200 GB on system
Networking Infrastructure	Information not provided.
Switching Technology	1HP4108XL – 100 Ethernet
Software	
Operating System and Version	RH 7.1
Compilers and Versions	Standard Red Hat 7.1 compilers. MPI is supported.
Batch System and Version	None (Each node requires Kerberos authentication and a batch system would have to be used that is able to use Kerberos to access the nodes)
Exclusive Priority Availability	None
Maximum Job Time (wall clock)	Not Applicable
Maximum Number of CPUs/Job	64 and 16, respectively for each of the GX240 and Workstation530 clusters.
Minimum Number of CPUs/Job	1

5.1.6.2 Informatics Beowulf Cluster Access Details and Support

University of Edinburgh Informatics Beowulf Cluster	
System Owner	University of Edinburgh, School of Informatics
System Administrator	University of Edinburgh, School of Informatics
System Availability Policies	Currently University of Edinburgh's Informatics related projects only.
System Maintenance Frequency	Irregular maintenance as required.
System Upgrade Schedule	Possible access to a new Network Attached Storage.
System Reliability (% uptime)	"Very Good" – quoted by Iain Rae.
Allocation Unit (AU) Cost	Not Applicable
CPU Hour Cost	Not Applicable
Available CPU Hours per Month	Not Applicable

5.1.7 The EPCC Microsoft Windows PC Cluster

The EPCC Microsoft Windows OS PC cluster consists of a set of desktop machines, which are the working machines of EPCC staff. This cluster is networked to a single Microsoft Windows OS server.

The number of available machines of each type and their details are listed in the table below. These details may change over time as the cluster evolves.

Number of Machines	PC Model	CPU (P – Intel Pentium) (Cel – Intel Celeron)	Memory	OS (SP – Service Package)
1	Dell - Dimension 4450	P4 – 2533	512 MB	XP SP1
1	Dell - Dimension 8100	P4 – 1500	640 MB	W2K SP4
1	Dell - Dimension 8100	P4 – 1500	1028 MB	W2K SP4
2	Dell - Dimension 8100	P4 – 1500	256 MB	W2K SP4
1	Dell - Dimension 8100	P4 – 1500	512 MB	W2K SP4
2	Dell - Dimension 8200	P4 – 1800	256 MB	W2K SP4
1	Dell - Dimension 8200	P4 – 1900	256 MB	W2K SP4
1	Dell - Optiplex E1 Celeron	Cel – 400	192 MB	W2K SP4
1	Dell - Optiplex GX1	P2 – 350	192 MB	W2K SP3
1	Dell - Optiplex GX110	Cel – 700	384 MB	W2K SP4
1	Dell - Optiplex GX110	P3 – 1000	384 MB	W2K SP4
1	Dell - Optiplex GX110	P3 – 667	192 MB	Test Machine
1	Dell - Optiplex GX110	P3 – 733	256 MB	W2K
3	Dell - Optiplex GX110	P3 – 866	256 MB	W2K SP4
1	Dell - Optiplex GX240	P4 – 1600	384 MB	W2K SP4
5	Dell - Optiplex GX240	P4 – 1700	640 MB	W2K SP4
1	Dell - Optiplex GX240	P4 – 1800	512 MB	W2K SP4
6	Dell - Optiplex GX260	P4 – 2400	512 MB	W2K SP4
3	Dell - Optiplex GX60	Cel – 2000	256 MB	XP SP1
1	Dell - Optiplex Gxa	P2 – 233	196 MB	W2K SP4
1	Dell - PowerEdge 2400	P3 - 667 Dual	256 MB	W2K Ser SP3
1	Dell - Precision			New - not yet built
1	Dell - Precision WS 330	P4 – 1700	1024 MB	W2K Ser SP3
1	Dell - Precision WS 610	P3 – 500	512 MB	Win NT

5.2 Discussion of the Rendering Computing Resources

The University of Edinburgh's Sun SMP cluster, Ness, is an attractive option to use for the rendering process. This system is administered by EPCC's system administration team who are based locally with respect to EPCC's PGPGrid project team. As well as its four processor front end, which allows interactive access, its 48 batch-processor back end offers good processing power. In addition, it has been expressed that exclusive access to the machine may be possible depending upon the requirements of the PGPGrid project. The determining factors for this are:

- the average wall clock time taken by jobs;
- the time of day and day of the week for running jobs;
- the number of jobs submitted at a time; and
- the amount of disk space required for the storage of output data.

In addition to Ness, the NeSC's Blue Dwarf IBM node and the ScotGRID Linux cluster have been offered for the PGPGrid project, although it has been stated that exclusive access to these machines is not possible and that their use will be subject to the load on the machines.

In order for all of these machines to be used concurrently in a parallel rendering environment on a Grid, rendering software must be found that can be run on all three platforms. That is, the rendering software must be available for:

- the Solaris 2.9/UltraSPARC III platform;
- the AIX 5.1/POWER4 System platform; and
- the Red Hat Linux 7.1/Pentium III platform.

This will be a challenging requirement to fulfil, although this solution adds value to the PGPGrid project in that it would demonstrate that the rendering process can be performed over a truly distributed and heterogeneous Grid of computing resources. In addition, it would prove that this process can be undertaken remotely with respect to PGP, making use of available Grid middleware technologies for data transfer, security, and resource management.

In the absence of suitable rendering software for the above three platforms, use will be made of the 3D Studio Max application's internal renderer. This software is only available for the Microsoft Windows OS platform, and thus, the EPCC Microsoft Windows OS PC cluster is the only suitable option for use as the rendering computing resource. Although this solution will not enable the PGPGrid project to demonstrate use of a distributed and heterogeneous Grid of computing resources for the rendering part of the project, it will nevertheless provide valuable experience with remote access to resources and with Grid middleware technologies for data transfer, security, and resource management.

6 Rendering Software

This section considers a host of different rendering software, giving technical details for each package, followed by some discussion and final recommendations. The possible rendering software include:

- Brazil from Splutterfish: <http://www.splutterfish.com>;
- Renderman from Pixar: <http://renderman.pixar.com>;
- Finalrender from Cebas: <http://www.finalrender.com>;
- The POV-Ray Freeware renderer: <http://www.povray.org>;
- Mental Ray from Mental Images: <http://www.mentalimages.com>; and
- 3D Studio Max from Discreet: <http://www.discreet.com>.

The following table includes the particulars of each of the rendering software in the list above.

Brazil (http://www.splutterfish.com)	
Supported Platforms	<ul style="list-style-type: none"> • Microsoft Windows 2000, NT, XP • 300 MHz Intel processor or better • 128 MB RAM & 300 MB swap or better
Requirements	Floating license available.
License Policy	<ul style="list-style-type: none"> • 4 node farm license – US \$750 • 3 node (1 × developer + 2 × farm nodes) – US \$1200
License Cost per Node	
Renderman (http://renderman.pixar.com)	
Supported Platforms	<ul style="list-style-type: none"> • Microsoft Windows (NT), 2000, XP • Silicon Graphics Iris, IRIX 6.2 and up • Intel/RedHat Linux 5.0 and up • Possibly Solaris; Pixar may have a Solaris farm.
Requirements	Not provided by vendor.
License Policy	Information regarding the number of machines a license covers were not provided by the vendor. It was stated, however, that a free academic license is available, which requires a £500 fee for initial set up and support.
License Cost per Node	US \$5000 – US \$8000 (depending on options) GB £3150 – GB £5000 (approximately)
Finalrender (http://www.finalrender.com)	
Supported Platforms	<ul style="list-style-type: none"> • Microsoft Windows 98, 2000, NT 4.0, XP • Must be able to run 3D Studio Max • Dual-processor recommended • 256 MB RAM or better
Requirements	
License Policy	Unknown at time of publication.
License Cost per Node	EUR €840 + 16% TAX GB £590 + 16% TAX (approximately)
POV-Ray (http://www.povray.org)	
Supported Platforms	<ul style="list-style-type: none"> • Microsoft Windows, at least Windows 95, • MAC OS/OSX, at least Mac OS 8.6 or Mac OS X 10.1.5, • i86 Linux, glibc-2.2 based GNU/Linux system
Requirements	(open source, can be compiled for running on other architectures) The same as the minimum required for the operating systems listed above, on each platform.
License Policy	None
License Cost per Node	Open Source Freeware
Mental Ray (http://www.mentalimages.com)	
Supported Platforms	Multiple processor arch/OS platforms, including <ul style="list-style-type: none"> • Microsoft Windows flavours, • UNIX flavours, • Linux
Requirements	Unknown at time of publication.
License Policy	Unknown at time of publication.
License Cost per Node	Available through resellers only.
3D Studio Max (http://www.discreet.com)	
Supported Platforms	Microsoft Windows
Requirements	<ul style="list-style-type: none"> • 300 MHz Intel or Athlon processor or better • 128 MB RAM & 300 MB swap or better
License Policy	License covers use for a farm of up to 9,999 machines.
License Cost per Node	Available through resellers only.

6.1 Discussion of the Rendering Software

The preferred renderers that have been specified as suitable by PGP are:

- Mental Images' Mental Ray;
- Splutterfish's Brazil;
- Pixar's Renderman; and
- 3D Studio Max's internal renderer.

The first three of these four rendering software packages are out-with the budget of this project. However, PGP's license for 3D Studio Max allows the use of its internal renderer at EPCC.

The POV-Ray Freeware renderer would also appear to be a contender, given the financial constraints on the project budget and its suitability to run on most of the resources outlined in Section 5.1. This choice would require that the available model data be translated into the POV-Ray data format. The model output available from PGP is in the 3D Studio Max binary format. There are a number of tools available for translating data from the 3D Studio Max binary format to the POV-Ray format, as listed in the Appendix: POV-Ray to 3D Studio Max Translators.

An additional advantage of using POV-Ray, is that not only can it be used on the preferred Ness SMP cluster for rendering, but due to its open source nature and its availability for multiple platforms, it will provide an opportunity to increase the range of rendering resources to also include the Blue Dwarf and/or the ScotGRID systems. This would provide a truly distributed and heterogeneous computing Grid for this project, and would demonstrate the capabilities of such a Grid for the rendering part of the project. As stated previously, the initial indications are that access to all three of these systems is possible. Since they are shared resources, however, PGPGrid jobs will run according to each system's fair share policies as defined by their administrators and as implemented by their batch systems, where available.

Following discussions with PGP, however, it is clear that the POV-Ray rendering software is not appropriate for their work. This is because the available translators are all primarily tools for exporting polygonal geometry to alternative rendering applications and are of very limited use in animation. The problem with these simpler exported formats, is that they do not support many of the sub-object deformations which have become so important in the 3D Studio Max format. Also, 3D Studio Max files are problematic in that they require the 3D Studio Max application itself to be present in order for them to be accessed. Further, 3D Studio Max compliant renderers are accessed from within the 3D Studio Max application (these renderers do not access the data files themselves for the rendering process). In this way, the 3D Studio Max application has overall control of the rendering process.

Our recommendation, therefore, is to employ the 3D Studio Max application's internal renderer. Although this solution lacks the ability to test the process of rendering on a heterogeneous Grid of computing resources, it does allow Grid middleware to be used to initiate and control the processes remotely. This is a feature that sets the project apart from existing render-outsourcing businesses, as it allows the animators to have full control of the rendering process. In addition, it may be shown that this solution is scalable to a larger number of resources upon demand, through the use of Grid middleware.

7 Recommendation for Overall Rendering Environment

The shortfalls of the POV-Ray rendering software for this project were highlighted by PGP and briefly described in Section 6.1. In addition, the limitations of the project budget do not allow for the acquisition of software licenses for the rendering software desired by PGP, and which are available for the Unix and Linux based platforms of the resources that are listed in Section 5.1.

The recommended rendering software, therefore, is the internal rendering engine of the 3D Studio Max animation framework. In this scheme, PGP would use their available 3D Studio Max licenses to undertake remote rendering on a cluster of EPCC desktop PCs running the Microsoft Windows OS, as described in Section 5.1.

Enabling rendering on a remote farm over a network will provide access to a plethora of resources. The Grid will bring this functionality for the PGPGrid project. Once remote rendering has been made possible on the EPCC desktop PC cluster, it could be possible to extend the set of resources to incorporate a range of PC clusters across the University of Edinburgh's campuses. These clusters will consist mainly of student computing laboratories.

Unfortunately, employing the 3D Studio Max application's internal renderer for the rendering environment, does not allow the PGPGrid project to investigate the possibilities of a heterogeneous Grid of computing resources for the rendering process. This will have to be investigated, if possible, by the ranging and modelling processes required for model production by 3D-Matic. This solution does, however, provide the opportunity to show that advantage may be taken of a large number of Grid enabled computing resources upon demand, through Grid middleware, and that this can be initiated remotely.

7.1 Recommendations for Further Investigation

Other possibilities that remain to be explored for the provision of a rendering environment for this project include:

- The use of Mental Images' Mental Ray rendering software on an AIX based platform such as the Blue Dwarf IBM node, that may in turn enable a single run of a set of rendering jobs on the HPCx supercomputer. This option depends on licensing issues being resolved between EPCC and Mental Images, in addition to a formula for access to the HPCx resource.
- The use of Splutterfish's Brazil rendering software on a Unix or Linux based platform, if porting to such a platform is allowed by Splutterfish and if licensing issues are resolved between EPCC and Splutterfish.

Appendix: POV-Ray to 3D Studio Max Translators

The following tools can be used to translate model data files from 3D Studio Max's binary format to the POV-Ray format:

- 3DS2POV, available at <http://www.personal.uni-jena.de/~p6sepa/software.htm> and <http://www.globserve.net/~snger/> ;
- AccuTrans 3D, available at <http://www.micromouse.ca/index.html> ;
- 3DWin, available at <http://www.tb-software.com/> ;
- NuGraf, available at <http://www.okino.com/> ; and
- Crossroads3D, available at <http://www.europa.com/~keithr/Crossroads/> .

These tools and others are available from:

- http://www.3dlinks.com/software_converters.cfm ; and
- http://www.povray.org/resources/links/3D_Programs/Conversion_Uutilities/ .

Acknowledgements

We would like to acknowledge the help of Josephine Beech-Brandt, John Fisher, and David Henty, all from EPCC, for providing information on the Ness SMP cluster. In addition, John Fisher provided information on the HPCx supercomputer and the HPCx development nodes. We also acknowledge the help of Craig Morris of EPCC for providing information on the EPCC Microsoft Windows OS PC cluster, David McNicol of the National e-Science Centre's system administration team for providing information on the Blue Dwarf IBM node, Tony Doyle of the National e-Science Centre for providing information on the ScotGRID Linux cluster, and Iain Rae of the University of Edinburgh's Informatics Division for providing information on their Beowulf cluster.

References

- [1] P. Cockshott *et al.*, Pepper's Ghost Productions Grid (PGPGrid) Project Proposal, National e-Science Centre Project Proposal, 2002.