ON DISTRIBUTED NETWORK RENDERING SYSTEMS

DEAN BRUTON
The School of Architecture, Landscape Architecture and Urban Design,
The University of Adelaide SA 5005 Australia
Dean.bruton@adelaide.edu.au

Abstract. This paper reports an investigation of the establishment and performance of a distributed computer rendering system for advanced computer graphics production within a centralized university information technology environment. It explores the proposal that the use of distributed computer rendering systems in industry and universities offers synergies for university-industry collaborative agreements. Claims that cluster computing and rendering systems are of benefit for computer graphics productions are to be tested within a standard higher education environment. A small scale distributed computer rendering system was set up to investigate the development of the optimum use of intranet and internet systems for computer generated feature film production and architectural visualisation. The work entailed using monitoring, comparative performance analysis and interviews with relevant stakeholders. The research provides important information for practitioners and the general public and heralds the initiation of a Centre for Visualization and Animation research within the School of Architecture, Landscape Architecture and Urban Design, University of Adelaide.

Keywords. Render farm, processing, computer graphics, animation.

1. Introduction

The establishment and performance of a distributed computer rendering system for advanced computer graphics production within a centralized university information technology environment may seem an easy task for those who have some experience in the IT industry. From an academic point of view (Bettis, 2005), it seems an attractive proposition because the idea that idle computers can be utilised for digital media production or for other intensive computing purposes makes economic and practical sense. The development of a render farm for to alleviate the tedious task of rendering thousands of frames for movies and architectural visualisations was enough reason to instigate an exploration of the potential synergies between university and industry in regard to collaborative arrangements for computer graphics production.

Many universities have added a digital media production element to either their schools of architecture, design or communication. The need for extended rendering power occurs towards the end of a semester when the student productions all need completion especially if there is a need to render ray traced materials or particle systems. The result is usually a desperate battle for computing time to use any computers that are available often overnight as well as during the day. The standard maximum time for a single frame for a movie is ideally 12 minutes. This is fine for a single image but a movie of any considerable length; say, 3.5 min (210 sec or 5250 frames) would be 63000 minutes or 43.75 full 24-hour days. If there is a class of 25 students all wanting to complete their work at the same time the pressure on the computers and personnel are intense. Krzyzanowski recognised the need for clusters of computers for benefits of reliability and scalability without the high cost" (Krzyzanowski, 2007).

Hence it should seem clear that there is a need to find ways to reduce the waiting and rendering time (The Google Code Channel, 2007), to utilise idle computers and to facilitate the development of a culture of excellence through regular reviews and critique of work. The latter only becomes available when the work can be readily and regularly rendered for reviews. The need for ongoing discourse regarding the development of a work is often regarded in industry as a key part of the production pipeline.

2. Background

For tertiary schools of animation and digital media production, the provision of distributed computer rendering systems assists staff and students and paves the way for a more professional postgraduate education in keeping with the demand of industry for graduates that are in tune with the practical requirements of the workplace environment. For production of high-quality CG-animated feature films such as *Astro Boy*, industry leader Peter Pang M.I.S. manager at Imagi recognises, "It is imperative we have in place a computing infrastructure that will meet the strong demands placed on our rendering application during production" (Business Wire, 2008).

Universities have a responsibility to prepare their graduates for the realistic demands of production environments once they leave the shelter of academic life. With this goal in mind a pilot project was devised to establish a small scale distributed rendering system to use computer aided architectural software together with plug ins that would normally require unacceptable conditions for production of rendered imagery in a traditonal computer lab setup without a renderfarm.

Students of the postgraduate course, Masters of Design in Digital Media use a computer laboratory of 25 seats for digital production for web, film and interactive media. The purchasing policy of the School allowed the reuse of older computers once they were replaced. The initial plan was to use 24 machines for the render farm. This was soon reduced as machines were repurposed and raided for parts to 16 machines. An air conditioned room was prepared to house the machines and the central IT Department began the render farm image set up process in liaison with the School's local IT officer, Ian Florance. The main issue to decide was the choice of software for the render farm operation.

3. Characteristics of Distributed rendering Systems

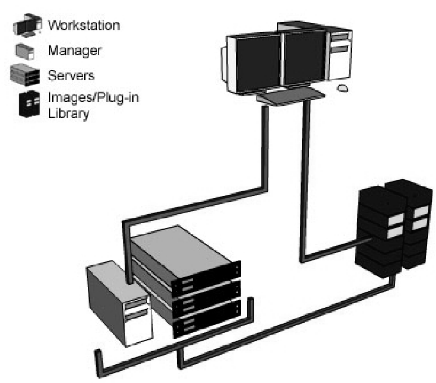

Figure 1. Typical render farm set up

A render farm is a computer cluster built to render computer-generated imagery (CGI), typically for film and television visual effects, using off-line batch processing. High performance computing has established a major research direction in Australia (Buyya, 1999). In contrast to the complexities of advanced grid computing, Matthew Tait describes the render farm set up process in detail and points out the pitfalls for this seemingly easy task: "A render farm employs many PCs simultaneously by running a queue manager on each box. The queuing software divides a job into multiple parts and decides which machine executes which part and when. Each machine refers to the job's scene file, which needs to reside in a location accessible to all the machines, and renders its share of frames. Once finished, each system stores the rendered frames back to that central location, ready for review." (Tait, 2005) Many researchers aim to design high-performance rendering systems by clustering a number of low-cost commodity PCs, each of which contains a commodity PC graphics accelerator. A typical configuration set up for a render farm is shown below:

There are several alternative queue manager software packages such as Dr Queue, Smedge, Qube!, Rush, Muster, Spider and 3Delight. After testing several candidates, we chose Qube! as it was suitable for a wide variety of applications and was an open-API and hence customisable. Using a Win XP operating system, the server on the network, called a Primary Domain Controller (PDC) or Open Directory Master (ODM), allows easy modification of passwords on the server, and you can tell another computer to refer to the server for login. A storage device is needed for the rendered imagery output. Connecting it to the PDC is a way of controlling access easily. Qube! requires a dedicated worker usually the slowest machine is used for this role. This machine tells other machines what to render when, but does no actual rendering itself. Once a baseline image is built on this machine, a client management tool (Ghost, Ghost For Linux, or OS X's Carbon Copy Cloner) is used to copy it to another hard drive. Install the supervisor application, in our case it was Qube! as this comes as a standard installer: MSI, tarball, or .dmg. Finally, once this is joined to the PDC's domain and the supervisor on the same hub as the PDC the first worker can be imaged. Select a machine to serve as your worker model.

4. The Pilot Project: Establishing a Rendering System in a Centralised Computing Environment

The central IT Department ensured Qube! was installed on the basic render farm image that was built for the local School IT Officer so that the configuration of Qube! could be completed. The central IT Department's role was to take the base student image and install the applications that were required (Qube!, 3dsMax) and put the machines on the network.

Basically it was like building a new student lab, as it created a new Organisation Unit in our Centralised University IT domain for the machines to sit in. Administration access was then given to the local School IT Officer. Installation was impeded by the Qube! staff not being able to directly access the link to assist. After several attempts to follow installation guidelines (Brew, 2007), install the software and test the prototype it became necessary to contact the software vendors for further advice.

In brief, the Qube! 5.2 installation began in June 2008. In July, a 3DS Max 2009 software upgrade caused further delay due to compatibility issues. In early November, testing was completed using the basic configuration with some software that exemplifies the computing requirements for commercial production. Results indicate the upgrade to Qube! 5.3 and a further year's license renewal are required for 2009 activities. Finally, in mid-December, the establishment of the render farm for testing purposes is finally under way with Qube! version 5.4.

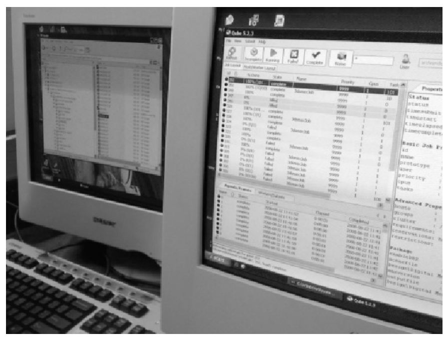

Figure 2. Qube! queue manager at University of Adelaide

5. Discussion

The establishment and use of a distributed rendering system or render farm in a higher education setting requires the assistance of both central and local IT services for installation and maintenance. Finding a balance between these entities that will result in fast and reliable results is problematic for a number of reasons. Firstly the need for ongoing communications with IT staff on a daily basis regarding the set up of a render farm proves to be difficult to achieve in reality due to the numerous responsibilities and duties that IT staff need to attend to. Academic staffs are limited in their ability to intervene in the technical installation process. Similarly they are dependent on the time frame and generosity of IT staff for an understanding of the best use of facilities and the correct process for render farm management. Issues such as the number and quality of machines, the acquisition of space and air-conditioning, the electrical wiring installation and the physical storage of machines are part of the ongoing concerns for all involved in render farm establishment.

5.1. BENEFIT OF DISTRIBUTED RENDERING SYSTEMS

The benefits of a distributed rendering system are:

- The faster rendering of imagery and movies
- The utilisation of idle computing resources
- The development of better working processes for the improvement of critique and output
- Cost savings in terms of time and management of files

These benefits need to be considered in the light of:

- Set up costs including ongoing license yearly fees
- Delays in liaison between centralised IT and local IT services
- Need for student understanding of the potential of working with a render farm
- Difficulties in expanding the use of computers within the education setting

For the render farm to be of value it was postulated that a Centre for Visualisation and Animation be part of the School's *modus operandi*. Using the distributed rending system was seen as part of a larger direction for the expansion of student movie production and the development of industry related activities such as collaborative work and shared projects. Due to the long delays in the set up and establishment of the render farm these plans were postponed until a satisfactory computer laboratory standard of delivery was finalised. This is expected to be completed for further testing in early 2009.

5.2 INTERVIEW WITH STAKEHOLDERS

Interviews were sought with other users of render farms throughout Australia. All 38 universities were contacted to develop a database of information about the use of render farms in higher education. Using the Pollograph website (*http://www.pollograph.com/ask/287/renderfarms-and-it/)* a survey was constructed for render farm managers and IT staff. The link and a request for information were sent to all universities by email during October 2008. Results were too limited to draw establish clear tends but some comments reveal major concerns such as, "to support render farms in a restricted financial climate".

Pascal Grosvenor, DAB Senior Computing Officer, UTS, Sydney recognises the importance of render farm provision in the final weeks of semester, "This is when students are working flat-out to get their final projects completed," Grosvenor explains. "And it's when the Xserves and Macs are being virtually punished 24 hours a day, seven days a week. ...With 25 frames per second and up to five minutes required for the final project, that's 7,500 individual frames that need to be rendered." (Apple, 2008).

In November-December 2008, interviews were held with staff involved with rendering in Australian universities including University of Sydney, University of NSW, University of Melbourne, RMIT, Swinburne, University of WA and Edith Cowan University. Staff comments on their experience of the installation and maintenance of special projects such as render farms were generally negative. Central IT services were considered by some to be "poison" because of (a) a loss of autonomy with the local computing environment (b) restrictions of online design and distribution access (c) financial implications for computing management. Many supervisors, (e.g. Bernard Mead) decided to use backburner for 3DS max as required. The need for both 3DS Max and Maya to use the render farm was addressed by users of Cube, but they found installation and support issues prevented a smooth setting up.

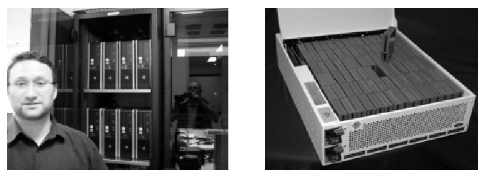

Figure 3. Bernard Mead, University of Melbourne; SGI Molecule Packs 2008.

Supercomputing groups were discusses as potential partners ion rendering for architectural and digital imagery projects, but it they offer rendering services on a fee for service basis or are too concerned with high mathematics research. University of WA used 2008 strategic grant funds to provide their own services using Dell as a support advisor. For Jamie Graham, using a minimum of two local technical support officers for their architectural and digital design School was considered essential. Graham found the installation of Cube took 5 days to configure correctly, and that backburner could supply a 3Ds Max rendering solution for their School. The need for both 3DS Max and Maya rendering was not part of their program.

Figure 4. Jamie Graham, University of WA with 2008 render farm (right).

The development of cloud computing is still to become part of university rendering environments but is now looking for acceptance in the industry. AMD in partnership with content developer OTOY are developing the fastest graphics supercomputer ever built by late 2009 (Hendrickson, 2000). The aim, Dirk Meyer President of AMD announced, is "to make a HD cloud computing a reality" (Jacobs, 2009). DreamWorks (Miller, 2009) are exploring this direction for real time rendering across a network.

However there are security issues with this option: "IBM argued that cloud computing is a way for businesses to draw more value from their existing IT infrastructures, since much of the work is offloaded onto remote servers, but the cloud-computing concept has received sharp criticism recently from the likes of Larry Ellison and Richard Stallman." (Broersma, 2008). For rendering within business system setups there are some fast internal solutions available (Vosburgh, 2009). This may currently be the easier path for University environments that are turning to networks for educational content delivery and management.

6. Conclusions

For many the dream of developing a powerful laboratory distributed network cluster rendering system is unachievable because the centralised University IT services are at odds with the needs of localised School digital media operations. This study has shown it is possible with determination to succeed by working through communication and technical barriers. Despite the hype (http://www.pipelinefx.com/) and consequent high expectations of the software publicity (Pipline fx, 2008) the reality of establishing a render farm proved difficult and frustrating within the higher education computing environment for many academics.

Support for the software was often delayed and incomplete, for example, we found the installation manual to be out of date. Major higher education institutions do not collaborate to share their experiences and rely on the newletters from software companies to guide their choice of technical support (SGI, 2007). We hope that the observations and data present in this paper help future designers of high-resolution or complex computing arrays in choosing data rendering strategies suitable for their system architectures. This work suggests several areas for future research:

Using resources more efficiently through collaboration of faculties and schools may be a future goal for digital design production laboratories. As the imperative for cost saving and output maximisation drives us to think of new ways of operating on reduced costs, collaboration and sharing of computer resources makes good sense especially in regard to software cost optimisation.

Another interesting issue for future work is the organisation of recycled hardware for distributed rendering systems. Too often higher education institutions discard machines on the so-called use by date. In an age where the earth's resources are scarce we need to find better methods of reusing the expensive hardware we too easily turn into waste.

The interdisciplinary approach offers alternative strategies for digital media production that contravenes conventions set when universities were based on strict disciplinary rules. Distributed rendering systems are one instance that strongly suggests we share resources across say, computer science, medicine, architecture and engineering etc when possible. Future solutions for distributed rendering are suggested by the SGI Molecule Packs 10,000 Atom Cores (Diaz, 2008).

The future provision of IT services for universities needs serious consideration because management issues become unwieldy once a critical mass of complexity is reached. Solutions such as the complete outsourcing of services for students and the provision of on demand providers indicate the need for a more flexible approach, with due consideration given to open source solutions. Many interviewees suggested that special project teams are needed to work with academics and local IT staff. The establishment and maintenance of a special project such as a render farm may be considered a test case for the health of an organisation's IT services infrastructure.

Acknowledgements

The author acknowledges the valuable ongoing contribution to this research of Ian Florance, and the IT Services of the University of Adelaide and to all interviewees around Australia.

References

Apple: 2008, Apple – Education, retrieved November 23, 2008, from Masters of Animation - Case Study: http://www.apple.com/au/education/profiles/uts/

Bettis, D.E.: 2005. Digital Production Pipelines: examining stuctures and methods in the computer effects industry. Thesis, Texas A&M University, Science, Texas.

Brew, S.: 2007, Qube! installation instructions. Adelaide, SA, Australia.

Broersma, M.: 2008, IBM joins the 'cloud computing' bandwagon, retrieved February 20, 2009, from Builderau: http://www.builderau.com.au/news/soa/IBM-joins-the-cloud-computing-bandwagon-/0,339028227,339292482,00.htm?feed=pt_saas

Business Wire: 2007, IBRIX Supports Imagi Studios' Expanded Production Pipeline, retrieved November 06, 2008, from Bnet: http://findarticles.com/p/articles/mi_m0EIN/ is_/ ai_n19040678

Buyya, R. (Ed.): 1999, High Performance Computing, Prentice Hall, New Jersey.

Diaz, J.: 2008, Computers, retrieved November 23, 2008, from SGI Molecule Packs, 10,000 Atom Cores: http://gizmodo.com/5091473/sgi-molecule-packs-10000-atom-cores-one

Hendrickson, M.: 2000, OTOY Developing Server Side 3D Rendering Techonology, retrieved February 20, 2009, from TechCrunch: http://www.techcrunch.com/2008/07/09/ otoy-developing-server-side-3d-rendering-technology/

Jacobs, D.: 2009, AMD to build fastest computer for... games, retrieved February 20, 2009, from Ibtimes: http://www.ibtimes.com/articles/20090108/amd-build-fastest-supercomputer-games_all.htm

Krzyzanowski, P.: 2007, Clusters; building scalable systems, retrieved November 6, 2008, from pk.org: Distributed Systems: http://www.cs.rutgers.edu/~pxk/rutgers/notes/content/clusters.html

Miller, R.: 2009, DreamWorks Eyes Render Farm in the Cloud, retrieved February 17, 2009, from Datacenterknowledge: http://www.datacenterknowledge.com/archives/2009/02/17/dreamworks-eyes-render-farm-in-the-cloud/

Morley, S.: 2002, *Realistic Ray Tracing*, AK Peters.

Pipeline fx: 2008, Rapid Adoption of Qube!TM For Education Continues by Universities and Schools Worldwide, retrieved November 6, 2008, from Pipelinefx News: http://www.pipelinefx.com/news/article.php?id=115

Samanta, R.E.: 2001, High Performance Parallel Rendering on a PC Cluster, retrieved from http://www.cs.princeton.edu/~rudro/cluster-rendering/

SGI.: 2007, Press release: SGI Technology Powers L'Ecole Gobelins Final Push for 31st Annual Annecy International Animation Festival, retrieved November 6, 2008, from SGI: http://www.sgi.com/company_info/newsroom/press_releases/2007/june/gobelins.html

Tait, M.: 2005, Build Your Own Renderfarm, retrieved November 7, 2008, from Extreme Tech: http://www.extremetech.com/article2/0,2845,1847411,00.asp

The Google Code Channel.: 2007, Cluster Computing and MapReduce Lecture 1, retrieved 12 7, 2008, from YouTube: http://www.youtube.com/watch?v=yjPBkvYh-ss&NR=1

Vosburgh, C.: 2009, Business Agility : Using internal cloud computing to create a computer animation render farm in less than a day, retrieved February 20, 2009, from Casstt Data: http://datacenterdialog.blogspot.com/2009/01/business-agility-using-internal-cloud.html