

## BORANG PENGESAHAN STATUS TESIS

JUDUL: IMPLEMENTATION OF NETWORK RENDERING FARM

SESI PENGAJIAN: 2009/2010

Saya SITI RASHIDAH BINTI MISHORIM @ MISROHIM  
(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Teknikal Malaysia Melaka
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. \*\* Sila tandakan (/)

\_\_\_\_\_ SULIT

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

\_\_\_\_\_ TERHAD

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/ badan di mana penyelidikan dijalankan)

\_\_\_/\_\_\_ TIDAK TERHAD

  
(TANDATANGAN PENULIS)

Alamat tetap: Lot 1556, Batu 3,  
Kg Sg Kandis, Seksyen 36,  
40460 Shah Alam,  
Selangor Darul Ehsan.

  
(TANDATANGAN PENYELIA)

Pn. Zaheera Binti Zainal Abidin

Tarikh : 24 JUNE 2010

Tarikh : 24 JUNE 2010

CATATAN: \* Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda (PSM)


\*\* Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

## DECLARATION

I hereby declare that this project report entitled

### IMPLEMENTATION OF NETWORK RENDERING FARM

is written by me and is my effort and that no part has been plagiarized  
without citations.

STUDENT :  Date: 24 JUNE 2010  
(SITI RASHIDAH BT MISHORIM@MISROHIM)

SUPERVISOR:  Date: 24 JUNE 2010  
(PN. ZAHEERA BT ZAINAL ABIDIN)

## **DEDICATION**

To my beloved parents, your love and support are my greatest and my motivation.

To my friends, for your support and encouragement.

To my lecturer, for being respective and critical, and challenging me to be better student.

## ACKNOWLEDGEMENTS

Alhamdulillah and thanks to Almighty Allah, I am very pleased and grateful of being able to finish my final project. First and foremost, I would like to thank my beloved parents and my family for their support and motivation throughout my project.

I would like to express my gratitude to my supervisor, Pn. Zaheera bt. Zainal Abidin, for her understanding and patience, added considerable to my success of completing this project. I appreciate her vast knowledge and skills in many areas and her guidance in writing and completing this report.

I also expressed my appreciations to my friends in and outside UTeM for their exchanges of knowledge, skills, and venting of frustration while completing my final project which helped enrich my experience. Although, I would like to thanks for many people that have contributed and helped to complete this project. I take sole responsibility for errors. Wassalam.

## ABSTRACT

Network Rendering Farm is one of the services in 3D rendering and animation in the network environment. It is a process of having multiple networked machines, working simultaneously, which the render server/master distributed the individual frames of an animation or 3D applications to number of render nodes/slaves, in reducing the amount of processing time. Network rendering farm use distributed rendering method because it allowed all render node to render the job. The frame that to be rendered is divided into small regions (buckets) and give each render nodes to render a number of them which then get the result and combine them into the final image. The final image result can be seen in render server/master. The network rendering farm service is implementing in IPv4 network environment. This service also uses different operating systems which is Ubuntu and Windows XP. Blender is used as software in render server to make 3D or animation. It is build with Visual C++ 2008 SP1 Redistribution Package (x64) and Python as script and language. Farmerjoe will be used also in render server as an engine for Blender software to send the jobs that ready to be rendering to the render node. Farmerjoe allowed many jobs to be sent for rendering and the jobs will start automatically after the other jobs are finished. As a result, the rendering time can be reduced by using more computers in network environment.

## Abstrak

*Network Rendering Farm* merupakan satu perkhidmatan 3D dan animasi dalam persekitaran rangkaian. Ia adalah proses yang mempunyai beberapa mesin rangkaian, bekerja secara serentak, yang mana *server / master* membahagikan *frame* individu aplikasi 3D atau animasi kepada *render node/slave*, dalam mengurangkan masa bagi proses *render*. *Network Rendering Farm* menggunakan kaedah pengagihan rendering kerana membolehkan semua *render node* untuk melakukan kerja. *Frame* yang akan di *render* dibahagikan menjadi kecil (*bucket*) dan diagihkan kepada setiap *render node* yang mana hasil akhir akan digabungkan dan boleh dilihat melalui *server/master*. Perkhidmatan ini dibangunkan di dalam rangkaian *IPv4*. Perkhidmatan ini juga menggunakan sistem pengoperasian yang berbeza iaitu *Ubuntu* dan *Windows XP*. *Blender* digunakan sebagai perisian dalam perkhidmatan ini. Ia turut dibangunkan beserta dengan skrip *Visual C++ 2008 SP1 Packege (x64)* dan *Python*. *Farmerjoe* juga turut digunakan sebagai mesin kepada perisian *Blender* untuk menghantar kerja yang sedia untuk di *render* kepada *render node*. *Farmerjoe* membolehkan penghantaran kerja secara banyak dilakukan dan akan bermula secara automatik setelah kerja yang terdahulu selesai. Kesimpulannya, masa bagi *rendering* dapat dikurangkan dengan menggunakan lebih banyak komputer dalam persekitaran rangkaian



## TABLE OF CONTENTS

CHAPTER	SUBJECT	PAGE
	DECLARATION	i
	DEDICATION	ii
	ACKNOWLEDGEMENTS	iii
	ABSTRACT	iv
	ABSTRAK	v
	TABLE OF CONTENTS	vi
	LIST OF TABLE	x
	LIST OF FIGURES	xi
CHAPTER I	INTRODUCTION	
	1.1 Project Background	1
	1.2 Problem Statement	2
	1.3 Objectives	3
	1.4 Scope	4
	1.5 Project Significance	5
	1.6 Expected Output	5
	1.7 Conclusion	6

## **CHAPTER II            LITERITURE RIVIEW AND PROJECT METHODOLOGY**

2.1 Introduction	7
2.2 Literature Review	7
2.2.1 Domain	7
2.2.2 Keyword	8
2.2.3 Previous Research	9
2.2.3.1 Down on the Farm (Network Rendering and Autodesk Backburner)	9
2.2.3.2 An Open Source Grid Based Render Farm for Blender 3D	11
2.3 Proposed Solution	15
2.3.1 Project Methodology	15
2.4 Project Schedule and Milestone	18
2.5 Conclusion	19

## **CHAPTER III            ANALYSIS**

3.1 Introduction	20
3.2 Problem Analysis	20
3.2.1 Network Architecture	21
3.2.2 Logical and Physical Design	21
3.2.2.1 Logical Design	22
3.2.2.2 Physical Design	23
3.3 Requirement Analysis	23
3.3.1 Quality of Data	24
3.3.1.1 Article	24
3.3.1.2 Software Requirements	24
3.3.1.2.1 Windows XP	24
3.3.1.2.2 Ubuntu	24



	3.3.1.2.3 Blender	25
	3.3.1.2.4 Farmerjoe	25
	3.3.1.2.5 Phytton	25
	3.3.1.2.6 Visual C++ 2008	26
	3.3.1.3 Hardware Requirement	27
	3.3.1.3.1 Cisco Switch 2690	27
	3.3.1.3.2 Cable	27
	3.3.1.4 Network Requirements	27
	3.4 Conclusion	28
<b>CHAPTER IV</b>	<b>DESIGN</b>	
	4.1 Introduction	29
	4.2 Possible Scenarios	29
	4.2.1 IPv4 Scenario	30
	4.3 Conclusion	31
<b>CHAPTER V</b>	<b>IMPLEMENTATION</b>	
	5.1 Introduction	32
	5.2 Network Configuration Management	33
	5.2.1 Configuration Environment Setup	33
	5.2.1.1 OS/Platform Configuration	33
	5.2.1.1.1 Windows XP	33
	5.2.1.1.2 Ubuntu	36
	5.2.1.2 Render Tools Configuration	37
	5.2.1.2.1 Blender	37
	5.2.1.2.2 Farmerjoe	38
	5.3 Hardware Configuration Management	42
	5.3.1 Hardware Setup	42

	5.4 Development Status	43
	5.5 Conclusion	43
<b>CHAPTER VI</b>	<b>TESTING</b>	
	6.1 Introduction	44
	6.2 Test Plan	44
	6.2.1 Test Organization	45
	6.2.2 Test Environment	45
	6.2.3 Test Schedule	45
	6.3 Test Strategy	46
	6.3.1 Classes of Test	47
	6.4 Test Design	47
	6.4.1 Test Description	48
	6.4.2 Test Data	49
	6.5 Test Result and Analysis	56
	6.6 Conclusion	57
<b>CHAPTER VII</b>	<b>PROJECT CONCLUSION</b>	
	7.1 Observation On Strengths and Weaknesses	58
	7.2 Propositions for Improvement	58
	7.3 Contribution	59
	7.4 Conclusion	59
	<b>REFERENCES</b>	60

## LIST OF TABLES

<b>TABLE</b>	<b>TITLE</b>	<b>PAGE</b>
Table 2.1	Comparison Between Different Available Render Farms	9
Table 2.2	Project Schedule	23
Table 5.1	Platforms IPv4 Network Address	23
Table 6.1	Test Schedule for Network Connection	31
Table 6.2	Test Schedule Render Server/Master and Render Nodes/Slaves	31
Table 6.3	Classes of Testing	32
Table 6.3	Test Network Connection between the Workstations	33
Table 6.5	Test Connection between the Render Server and Render Nodes	33
Table 6.6	Network Rendering Farm Work	33
Table 6.7	Network Rendering Farm Performance	34

## LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.1	Architecture of the Open Source Blender Render	8
Figure 2.2	Rendering Improvement of Grid over Single Computer	11
Figure 2.3	The Process Model for Requirement Analysis	18
Figure 3.1	Logical Design	20
Figure 3.2	Physical Design	27
Figure 4.1	IPv4 Scenario	28
Figure 5.1	Workgroup Interface	29
Figure 5.2	Map Network Drive Interface	30
Figure 5.3	Shared Directories	
Figure 5.4	Shared Directories (R: Server\RenderFarm)	36
Figure 5.5	Samba Configuration	37
Figure 5.6	Blender Interface	38
Figure 5.7	Farmerjoe Configuration	39
Figure 5.8	Farmerjoe_submit and Farmerjoe_bucket	40
Figure 5.9	Master Command	41
Figure 5.10	Appserver Command	46
Figure 5.11	Slave Command in Windows XP	47
Figure 5.12	Slave Command in Ubuntu	48
Figure 5.13	Farmerjoe Interface	49
Figure 5.14	Hardware Setup	50
Figure 6.1	Render Server ping to Render Nodes	51
Figure 6.2	Render Nodes ping to Render Server	52
Figure 6.3	Render Server/Master in Windows XP	53
Figure 6.4	Render Appserver in Windows XP	

- Figure 6.5**      **Render Nodes/Slaves in Ubuntu**
- Figure 6.6**      **Render Nodes/Slaves in Windows XP**
- Figure 6.7**      **Farmerjoe Interface in the Web Browser**
- Figure 6.8**      **Send Job by using Blender**
- Figure 6.9**      **Rendering Process by Render Nodes**
- Figure 6.10**     **Jobs Status**
- Figure 6.11**     **Jobs Complete**
- Figure 6.12**     **Results of Network Rendering Farm Time Comparison  
with Number of Computers**

## **CHAPTER I**

### **INTRODUCTION**

#### **1.1 Project Background**

Rendering is the process that involves in multimedia especially in animation and 3D. It is usually take a long time of processing to finish because it is done in one computer only. Nowadays, the rendering process can be done in multiple nodes and it is known as network rendering farm. The network rendering farm is the process of having multiple networked machines to render the individual frames of an animation in order to reduce the amount of time needed. There is render server or master and render node which also known as slave that involves in network rendering farm.

Network rendering farm use distributed rendering method because it allowed all render node to render the job. The frame that to be rendered is divided into small regions (buckets) and give each render node to render a number of them which then get the result and combine them into the final image. The final image result can be seen in render server/master.



divided the frame into a small to be rendered by the render node. It is allowed to check the frame which already rendered to see its progressing while the other still going on. If there any errors occurred, the job can be terminated and resubmit again. While waiting the rendering finished, another job can be done simultaneously.

### 1.3 Objective

Currently, the rendering process not been implemented in network environment. It processed only in one machine. The objectives to implement network rendering farm services are:

- i. Provide the services in network environment by implement and apply the services using IPv4 addresses.
- ii. Upgrade and increase the productivity and performance of rendering process.
- iii. Achieve a multiple simultaneous rendering frame rate.
- iv. Established the network rendering farm in different platforms which are Ubuntu and Windows XP.
- v. Make the server node and the render node communicate successful.

The network rendering farm service is implementing in IPv4 environment. In fact, this service is implemented in different platform which is Ubuntu and Windows XP. This project ensures that the services worked and the communication between the render server and render node is successful.

## **1.2 Problem Statement**

The rendering process usually is done in only one computer. Basically, the only one computer act as platform which use to make 3D and animation process and also act as a machine to make a rendering process. This usually takes a long time to complete and make the performance become slow.

There is only one job can be rendered in one time. After one job is finish, it is needed to be close to start or load another job to be render. It is done manually. The results of the job can be seen after the project is finished rendering process. If there is an error occurred, it is only can be know at the end of the process. To fix the error, the job needs to be rendering again.

The network rendering farm is use to solve the problems that happened before. It is implemented in network environment with more than one computer which have render server and render node to do the rendering process. The 3D and animation process only involve in render server while the rendering process is done by the render node. This made the performance become faster and reduce the rendering time.

Moreover, this service allowed many jobs to be rendering in a time. The new job is start automatically after one job is finished. This service also has render queue which

## 1.4 Scope

The scopes involved in this project are:

- This project is carried out in the lab where the service is tested in real-environment using three workstations as server and a client that connected to each other using a Cisco switch and connected to a Cisco router in the same network.
- For the render server, it runs in Windows XP Operating System. The server contains software Blender-2.49b which includes Python 2.6 and Visual C++ 2008 SP1 Redistribution Package (x64). To render the job, the server needs Farmerjoe\_0.1.3.
- For the render node, it runs in Windows XP and Ubuntu 9.04 Operating System. The nodes also contain software Blender-2.49b which is the configuration is same as the render server. To receive the job from the server, the nodes copied Farmerjoe\_0.1.3 configuration in the server.
- The communication between the render server and the render node is using the Cisco Switch series 2960 in the same network environment.

## **1.5 Project Significance**

Network rendering farm is a service that helped the 3D rendering and animation render process to be more easy and faster on the performances. It reduced the rendering time process and allows many jobs in time. From this service, user which normally people who involved in multimedia can save their time and make another animation or 3D.

This service is easy to use because user just needs to send their job to the server node and the server divided the job to the render node that in the same network. The complete job sent back to the server and user can view it to see the results.

The service also efficient where it allows many jobs in a time and queue the job to be render after the other job is complete. Besides, the user also can make the job that is in the rendering process to be in the paused condition if it necessary.

## **1.6 Expected Output**

As an expected output, the network rendering farm service worked successfully so that the rendering time process can be reduced and give the best final result. The network rendering farm service was more efficient if it can be use to render many jobs in a time without requiring user to spend their time waiting for rendering process is completed.

The network rendering farm also was successfully communicate in different platforms which is Ubuntu and Windows XP that has been implemented in same network environment by using IPv4 addresses.

### **1.7 Conclusion**

This project is mainly about implementation of network rendering farm in network environment. This project was undertaken using three pc's, one as server and the other two as clients connected to each other using a switch. As the project carry out, comparison data was gather to determine that the network rendering farm is better than the old service. In the next chapter, it covered about the literature review and project methodology where it discussed about domain, keyword, previous research and project schedules and milestones.

## **CHAPTER II**

### **LITERATURE REVIEW AND PROJECT METHODOLOGY**

#### **2.1 Introduction**

In this chapter, both literature review and project methodology is discussed thoroughly for the project. Some journals, books and articles either in written form or online source is used as a cited overview about the current research for this project. As for the project methodology, a selected methodology was chosen based on the previous research and is carried out for the rest of the project activities until completion where it includes the techniques, hardware and software requirements that are used to develop the project.

#### **2.2 Literature Review**

##### **2.2.1 Domain**

Domain is the space or thing which is filled with some main features. In this project, domains that are related to this through the research and reference are network rendering farm, Farmerjoe\_0.1.3, Blender-2.49 and IPv4 addresses.



### 2.2.2 Keyword

There are few terms that being used in this project which are:

- I. Ipv4 Addressing - The IPv4 address is a 32-bit binary address represented as dotted decimal where the decimal values ranging from 0-255.
- II. Network rendering farm – One of the services where it main purposes is to speed up the rendering process in network environment.
- III. FarmerJoe\_0.1.3 – is a render farm implementation based on TCP/ IP connections to render Blender 3D models, it provides web based interface to schedule the jobs also the functionalities of Bucket based and frame based rendering facilities.
- IV. Blender-2.49 - is an integrated application that enables the creation of a broad range of 2D and 3D content. Blender provides a broad spectrum of modeling, texturing, lighting, animation and video post-processing functionality in one package. Blender is one of the most popular Open Source 3D graphics applications in the world.
- V. Ubuntu (9.04) - Ubuntu is a community developed operating system that is perfect for laptops, desktops and servers. Whether you use it at home, at school or at work Ubuntu contains all the applications you'll ever need, from word processing and email applications, to web server software and programming tools.
- VI. Windows XP - Windows XP is an operating system produced by Microsoft for use on personal computers, including home and business desktops, laptops, and media centers. It was released in 2001. The name "XP" is short for "eXPerience."

### **2.2.3 Previous Research**

A network rendering farm is a computer cluster built to render computer-generated imagery (CGI), typically for film and television visual effects, using off-line batch processing. Render farm is a service that involved in multimedia especially in 3D Renderings and Animations. Normally, the time for 'rendering' take too long to complete. This is because there is no network rendering is been using before.

Basically in the old concept of rendering only use a single computer to complete only one work of rendering complex animations. It may probably take many days to render single animation only. It also can only just complete one work of 3D and animations while there are still many works to do.

#### **2.2.3.1 Down on the Farm (Network Rendering and Autodesk Backburner)**

According to Gary M.Davis (2003, 2005 and 2006), the network rendering had been use and the new software called 'Backburner' was introduced on the mid of 2002. With the purchase of one copy or license, user can legally network render on up to 9,999 computers and they can be multi-processor. This means that one work can be handling by many computers. One computer takes charge on tasks for the rest of the computers to work on which is often called as a render farm. Besides, one computer started rendering a frame while the other can start to render another frame. This method of network rendering is known as distributed rendering which is the way of Backburner processes animation and composites.

There are many advantages of using and applying network rendering by using Backburner software which are:

- I. The Render Queue – the biggest advantages of network rendering is render queue concept which is it allows to submit jobs and let the jobs manage by its own self. It is means that when one job had finish, the next will begin automatically. It is different if there is no network rendering had been use. Without a queue, it is needed to close the project that finished and create or load another to begin rendering the next one.
- II. Revision – it is mean that the network rendering allow more version of the same animation in the same amount of time to be process. This will help to know and choose the best result for the frame. Besides, there only take a few times to complete the rendering process.
- III. Sequential Files – network render allow frame to be arrange starting from zero. It helps the frame render in sequence. Besides, we can check and open the frame that already rendered to see its progressing while the other still going on. If there are any errors, the rendering process can be terminated and can be fixes out to resubmit the job again. There is no need to wait for the entire job to finish seeing the errors.
- IV. Monitors and Logs – the rendering process can be monitored anywhere in the facility and diagnose its results, progress and shortcomings. It also helps in finding which computer is faster to render and can maximize the render farm's efficiency. While the render log allow to identify which computer rendered bad frames.

### **2.2.3.2 An Open Source Grid Based Render Farm for Blender 3D**

Blender 3D is an open source 3D authoring tool which use in the implementation of render farm management tool. Such an approach, it is very useful for embedding within a specific application area. The final system is about to move towards service orientation architecture where the services offered by three components which is Blender 3D, render farm management tool and Condor Grid. The Grid based render farm system includes set of functionalities and unique features which are not currently available in other solutions.

The aim for this project also to develop a 3D collaborative environment based on Grid and P2P infrastructure where user can collaboratively work on same 3D model. Usually, 3D artist can share their same graphics file and work in parallel to design a common model and animation when they are finished the stage of their job and can submit the jobs for rendering on the Grid, eliminate the need for dedicated render farm server.

Currently, there is a lot of research in the area of render farms for 3D applications which can be seen in the Table 2.1 below. Table 2.1 presents some of useful features which make the work different from others. These features are important as they allow the flexibility of deployment and cost effectiveness. An open source Grid based render farm tool that can be deployed on many platforms is very cost effective to end user.