KERNEL RE-IMPLEMENTATION

A PROJECT REPORT

Submitted By

VIKRAM TIWARI
(0901010118)

Shivam Gupta
(0901013080)

Tanishka Chaturvedi
(0901044061)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

Computer Science and Engineering

United College of Engineering and Research, Allahabad

Gautam Buddha Technical University, Lucknow

APRIL 2013

CERTIFICATE

This is to certify that the project report entitled "**KERNEL RE-IMPLEMENTATION**" presented by **VIKRAM TIWARI (0901010118), SHIVAM GUPTA (0901013080), and TANISHKA CHATURVEDI (0901044061)**, submitted in the partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science and Engineering of the Gautam Buddha Technical University, Lucknow during the academic year 2012-13, is a bonafide record of work carried out under my guidance and supervision.

Mr. Sanjay Pandey
Head of Department, Department of Computer Science
United College of Engineering & Research, Allahabad

ACKNOWLEDGEMENT

We express our sincere gratitude to **Mr. Sanjay Pandey**, Head of Department (B-Tech), Computer Science and Engineering Department for giving us an opportunity to work on this project. Under his guidance, constructive suggestions and all kinds of possible mentorship extended throughout the project the possibilities were endless. It's a nice working experience under his guidance as a team and would be very beneficial for us in the future.

Thanks are also due to all those who directly or indirectly extended their co-operation towards the project in any form.

----------	----------	----------
Vikram Tiwari 0901010118	Shivam Gupta 0901013080	Tanishka Chaturvedi 0901044061

ABSTRACT

Introduction:

The kernel is the program that performs the most basic functions of an Operating System: It controls the interfaces with the computer's hardware, handles allocation of memory and other resources, allows multiple programs to run at same time, manage the file system , and so on.

Objective:

The project aims at developing a Micro Kernel that provides the core functionality for the Operating System. Using this kernel we can interface with a wide variety of hardware components and abstract the functionality to system utilities, windowing systems and graphical desktops resulting in a fully-fledged operating system.

Scope:

This project will provide a platform that will provide the interface to the variety of hardware devices and with the use of system software and GUI components it can also provide a complete Graphical operating system. The key focus of our project is the applicability and variability of access and use.

With a completely parallel programming approach we can provide the multicore processing functionality and process intercommunication. The kernel will also provide a basic interfaced building block for the overall functionality of the system.

Workflow:

Vikram Tiwari: Team Leader and Platform maintainer

Shivam Gupta: Device and Algorithms maintainer

Tanishka Chaturvedi: Testing and Bug reporting maintainer

Contents

List of Figures……vii

Chapter 1 Introduction……1
1.1 An Overview……1
1.2 Objectives of the Project……2

Chapter 2 System Analysis……3
2.1 Proposed System……3
2.1.1 Problem Definition……3
2.2 System Design……4
2.2.1 Hardware Interfaces……4
2.2.2 Logical Interfaces……5
2.2.3 Bridges……5
2.2.4 Virtual File System……5
2.2.5 User Mode……6
2.3 System Specifications……7
2.3.1 Kernel Module Specification……7
2.3.1.1 Kernel Components……7
2.3.1.2 Integration Design……8
2.3.2 Environment Setup……9
2.3.2.1 Base System……9
2.3.2.2 Compiling……9
2.3.2.3 Useful Scripts……9
2.3.3 Hardware Specification……10
2.3.4 Software Specification……10

Chapter 3 Testing……11
3.1 Pre-Development Testing……11
3.1.1 Linux Kernel Testing……11
3.1.2 Testing Strategies……11
3.1.3 Testing Methods……12
3.2 Kernel Internals……12
3.2 Conclusions……14

Chapter 4 Design and Development Process……15
4.1 Fundamental Design Concepts……15
4.1.1 Fundamental Design Concepts……15
4.1.2 Modularity……15
4.1.3 Software Architecture……15
4.1.4 Structural Partitioning……15
4.1.5 Data Structure……15
4.1.6 Software Procedure……16
4.2 System Design……16
4.2.1 Data Flow Diagram……16

4.2.2 ER Diagram……..18
4.3 Design Process…….18
4.3.1 Boot Code……18
4.3.2 Input Design…….19
4.3.3 Compiling, Linking and Running……19
4.3.4 Output Design……..20
Chapter 5 Scope for Further Enhancements……21
Chapter 6 References……...22

List of Figures

Figure 1.1 – Basic Kernel Structure……..1

Figure 2.1 – Bare System Response on Boot Up……..4

Figure 2.2 – Initializing the system with Boot Loader……..4

Figure 2.3 – First Directed Response from the System…….5

Figure 2.4 – Interrupting the System Run…….5

Figure 2.5 – Clocked Interrupt via Interrupt Request Queue……....5

Figure 2.6 – Virtual File System Showing Directories and File Contents…….6

Figure 2.7 – Multiple tasks running in Fork processes……..6

Figure 2.8 – Higher Level Abstraction of a Kernel……...7

Figure 3.1 – Directory Structure of Linux Kernel……...11

Figure 3.2 – Kernel Diagram on Functional Layer View……13

Figure 3.3 – Kernel Diagram Detailing Calling Methods……14

Figure 4.1 – High level Kernel Modules Distribution…….17

Figure 4.2 – Memory and Virtual Space Allocation……18