LINUX CLUSTER MANAGEMENT TOOL

By

A.RAJESH

Reg. No 71202621031

of

Kumaraguru College of Technology

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

June, 2005

BONAFIDE CERTIFICATE

Certified that this project report titled

LINUX CLUSTER MANAGEMENT TOOL

Is Bonafied work of

Mr. A.RAJESH (Reg. No: 71202621031)

Who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

18/5/05

Project Guide

Head of the Department

We examined the Candidate with University Register No. 71202621031

in the project Viva-Voce examination held on 24/06/05

Internal Examiner

24/5/05

External Examiner

Inspirations systems

7/135 Head Quarters Road,
Uppilipalayam (Opp. Immanual Church)
COIMBATORE - 641 018. Cell: 936 326 2233
Telefax: 0422 - 539 3789

TO WHOMSOEVER IT MAY CONCERN

May 12, 2005

This is to certify that Mr.A.Rajesh (02MCA32) of Kumaraguru College of Technology has undergone his project "Linux Cluster Management Tool" in this organization from December 2004 to May 2005.

Due to proprietary reasons, the candidate is not allowed to use source code and design information in any form outside INSPIRATIONS SYSTEMS premises. Also the technical ideas of this project should be strictly used only for academic purpose with the prior permission from INSPIRATIONS SYSTEMS, Coimbatore.

We are pleased to certify that he has successfully completed his project on time to meet our requirements. During the period of the project work his conduct was good and we wish him all success in future endeavours.

Yours Truly,

J.RAJU

(For Inspirations Systems)

ABSTRACT

Most super computers in the world are built on the concept of parallel processing—high-speed computer power is achieved by pulling the power from each individual computer.

Commercial supercomputers are not affordable for all organizations because of the cost factor. It is not easily affordable for organizations to replace or upgrade an existing supercomputer with another one as computational need increases. In order to solve the above problems clusters are developed.

Cluster is a cheap and easy way to take off-the-shelf components and combine them into a single supercomputer. In the last several years, many universities world-wide have set up clusters for the purpose of scientific research or simply for exploration of the frontier of super computer building.

Clusters are surprisingly powerful and it is possible to get computational performance needed without committing millions of dollars to supercomputers. Clusters have the distinct advantage in that they are very simple to build using components available from hundreds, if not thousands, of sources. So there is no need to use new equipment to build a cluster and can be built from commodity hardware.

Cluster unlike supercomputers are independent of a single vendor or single source for equipment. The main advantages of cluster are low cost, easy up gradation and efficient maintenance.

Since a cluster is a collection of individual nodes, it is difficult for the administrator to manage each node individually. So a tool is necessary to manage the entire cluster from the server node.

The **Linux Cluster Management Tool** is developed for the cluster administrator to administer the slave nodes easily from the master node.

The Linux Cluster Management tool consists of

- Server Module
- Client Module
- Database Module

Server Module

The server module will run in the master node and is accessible only by the Administrator. The server module checks for the availability of the node requested by the administrator. This is achieved by sending IP address and port number to the client system. If the specific system has active port the connection is established.

Client Module

The client module is run in slave nodes. The client module is automatically started when the system is booted. It continuously listens for the servers request through a specific port. When it receives a request from the server it sends the performance details to the server. It also monitors the client activities and all unauthorized access is reported to the server.

Database Module

The database module is run in the master node of the cluster and is accessible only by the administrator. The nod and the user administration is done by the administrator.

ACKNOWLEDGEMENT

At this pleasing moment of having successfully completed the project work, I wish to acknowledge my sincere gratitude and heartfelt thanks to our beloved Principal **Dr.K.K.Padmanabhan** for having given me the adequate support and opportunity for completing this project work successfully.

I express my sincere thanks to **Prof. Dr. S.Thangasamy,** the ever active and sympathetic, Head of the Department of Computer science & Engineering, who with his careful supervision has ensured me in attaining perfection of work.

I extend my sincere thanks to **Asst. Prof. Mr. A. MuthuKumar M.Sc, MCA, M.Phil,** Course Coordinator of Computer Applications for rendering us all the timely helps through out the project.

I regard my heartfelt thanks and everlasting gratitude to my Project Guide **Ms.P.Parameswari MCA, M.Phil,** Lecturer, Department of Computer Applications for her uplifting ideas, inspiring guidance and valuable suggestions, which have been very helpful in refining upon the project.

Table of Contents

CHAPTER 1

INTRODUCTION

1.1 Organization Profile

Inspirations Systems is primarily a system and network integration company with its prime focus on Network design, implementation and training. The company was promoted in the year 2003. The company specializes as a network service provider covering a broad band spectrum of technologies. The company has considerable experience in delivering custom solutions, and providing network services to leading corporate companies and business enterprises.

Inspirations Systems have an experienced team of Networking Consultants to provide networking solutions. Inspirations Systems provide Network solutions for real networks in all major environment (Linux, Novell, Microsoft, etc.) for small and medium scale enterprises and computer solutions for end users and Corporate Company. Inspiration Systems provides custom built training as per the Client's specification.

Inspirations systems provides the following expertise to help the clients realize their objectives.

- Vertical industry domain knowledge.
- State of the art technology expertise.
- Business transformation core competency.

- Highly disciplined program management & change management competence.
- Organizational agility
- High end Networking solutions

1.2 SYSTEM ENVIRONMENT

HARDWARE CONFIGURATION

Server Configuration

Processor : Intel Pentium 4 Processor 2.8 GHz

Hard Disk : 80 GB

RAM : 512 MB

NIC : Two 10Mbit/second Ethernet adapters

Client Configuration

1st node

Processor : Intel Pentium 2 Processor

Hard Disk : 20 GB

RAM : 256 MB

NIC : Two 10Mbit/second Ethernet adapters

2nd node

Processor : Intel Pentium 3 Processor

Hard Disk : 20 GB

RAM : 256 MB

NIC : Two 10Mbit/second Ethernet adapters

1.3 SOFTWARE CONFIGURATION

Development Platform: Linux

Deployment Platform : Linux

Scripting Language : PHP v5.0 (For Database Access)

JavaScript (For Dynamic Design)

Database : MySQL v4.1.7

Web Server : Apache 2.0

Browsers : Mozilla Firefox 1.0

1.4 TECHNOLOGY USED

Linux Cluster Management tool is a web-based system with database access, so different languages are used for coding. The languages used and the reasons are described briefly below.

JAVA v1.5

Java is a dynamic and most powerful language in today's demanding programming field. The buzzwords that make Java happen are Simple, Secure, Portable, Object-oriented, Robust, Multithreaded, Architectural-neutral, Interpreted, High performance, Distributed and Dynamic that was conceived by James Gosling, Patrick Naughton, Chris Warth, Ed Frank and Mike Sheridan at Sun Microsystems, Inc. in 1991. The language was initially called "Oak" but renamed in 1995. Java language is developed for two most important forms- Application programming and Web Programming.

Java's architecture arises out of four distinct but interrelated technologies:

- The Java programming language
- The Java class file format
- The Java Application programming Interface
- The Java virtual machine

The key that allows Java to solve both security and the portability problems just described is that the output of a Java compiler is not executable code. Rather, it is bytecode. Bytecode is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for bytecode. Because the execution of every Java program is under the control of the JVM, the JVM can contain the program and prevent it from generating side effects outside of the system.

Multithreading is a more sophisticated method of running multiple instances of a single program simultaneously. Unlike C and C++, Java has a sophisticated set of synchronization primitives that are based on the widely used monitor and condition variable paradigm introduced by C.A.R. Hoare. By

integrating these concepts into the language (rather than only in classes) they become much easier to use and are more robust.

PHP v5.0

PHP, recursive acronym for "PHP: Hypertext Preprocessor" is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. PHP is a powerful and flexible tool. This power and flexibility comes from PHP being a very thin framework sitting on top of dozens of distinct 3rd-party libraries. Its syntax draws upon C, Java, and Perl, and is easy to learn. The main goal of the language is to allow web developers to write dynamically generated WebPages quickly. Instead of writing a program with lots of commands to output HTML, you write an HTML script with some embedded PHP code to do anything. What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server. The best things in using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer.

PHP is mainly focused on server-side scripting, so it can do anything like any other CGI program can do, such as collect form data, generate dynamic page content, or send and receive cookies.

There are three main fields where PHP scripts are used.

- Server-side scripting.
- Command line scripting.
- Writing client-side GUI applications.

PHP can be used on all major operating systems, including Linux, many Unix variants (including HP-UX, Solaris and OpenBSD), Microsoft Windows, Mac OS X, RISC OS, and probably others. PHP has also support for most of the web

servers today. So with PHP, we have the freedom of choosing an operating system and a web server.

One of the strongest and most significant features in PHP is its support for a wide range of databases. Writing a database-enabled web page is incredibly simple. We also have a DBX database abstraction extension allowing you to transparently use any database supported by that extension. Additionally PHP supports ODBC, the Open Database Connection standard, so we can connect to any other database supporting this world standard. Furthermore, we also have the choice of using procedural programming or object oriented programming, or a mixture of them.

PHP also has support for talking to other services using protocols such as LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (on Windows) and countless others. we can also open raw network sockets and interact using any other protocol. PHP has extremely useful text processing features, from the POSIX Extended or Perl regular expressions to parsing XML documents. For parsing and accessing XML documents, it supports the SAX and DOM standards. We can use our XSLT extension to transform XML documents.

MySQL v4.1.7

The MySQL software delivers a very fast, multi-threaded, multi-user, and robust SQL database server. MySQL Server is intended for mission-critical, heavy-load production systems as well as for embedding into mass-deployed software. MySQL is a registered trademark of MySQL AB. The MySQL software is Dual Licensed. Users can choose to use the MySQL software as an Open Source/Free Software product under the terms of the GNU General Public License.

MySQL is a relational database management system that stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of ``MySQL" stands for ``Structured Query Language." SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard.

MySQL software is Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License).

MySQL Database Server is very fast, reliable, and easy to use. It also has a practical set of features developed in close cooperation with its users. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

MySQL Server works in client/server or embedded systems. It consists of a multi-threaded SQL server that supports different backend, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). It also has different types of storage engines such as Heap, InnoDB, MyISAM, etc.

MySQL had a 4GB (4 gigabyte) limit on table size. With the MyISAM storage engine in MySQL 3.23, the maximum table size was increased to 8 million terabytes (2 ^ 63 bytes). With this larger allowed table size, the maximum effective table size for MySQL databases now usually is determined by operating system constraints on file sizes, not by MySQL internal limits. The InnoDB

storage engine maintains InnoDB tables within a tablespace that can be created from several files. This allows a table to exceed the maximum individual file size. The tablespace can include raw disk partitions, which allows extremely large tables. The maximum tablespace size is 64TB.

JavaScript v1.3

JavaScript is a JavaScript is a cross-platform, object-based scripting language developed by Netscape to enable Web authors to design interactive sites. Although it shares many of the features and structures of the full Java language, it was developed independently. JavaScript can interact with HTML source code, enabling Web authors to spice up their sites with dynamic content. JavaScript is endorsed by a number of software companies and is an open language that anyone can use without purchasing a license. It is supported by recent browsers from Netscape and Microsoft, though Internet Explorer supports only a subset, which Microsoft calls Jscript.

It can be imbedded in Web pages and read by the browser. It can be used to do things such as open a separate browser window or to display a message when the mouse moves over an object on the page. With JavaScript, you have many possibilities for enhancing your Web page with interesting elements. It allows the creation of dynamic content on Web pages.

JavaScript is often used for creating functions to automatically change a formatted date on a Web page, cause a linked-to page to appear in a popup window, cause text or a graphic image to change during a mouse rollover. Whether JavaScript can be run properly highly depends on what browser clients use and it is independent of what hosting provider's server platform.

CHAPTER 2

SYSTEM REQUIREMENT AND SPECIFICATION

The Software Requirements Specification is a technical specification of requirements for the software product. The goal of software requirements definition is to completely and consistently specify the technical requirements for the software products in a concise and unambiguous manner.

The Software Requirements Specification is based on the system definition, high-level requirements specified during initial planning. The requirement specification is primarily concerned with functional and performance aspect of the software product and emphasis is placed on specifying product characteristics without implying how the product will provide those characteristics.

Desirable properties of a Software Requirement Specification

- Correct
- Complete
- Consistent
- Unambiguous
- Functional
- Verifiable
- Traceable.

In this chapter, a deep analysis of the system requirements is provided. It also describes the modules involved in the system development. The development methodology followed to develop the system is then given.

2.1 EXISTING SYSTEM

Generally high-level computation speed can be achieved by the use of commercial supercomputers.

The disadvantages of the system are

1. Supercomputers are costlier.
2. It is difficult to upgrade when necessary.

There is no Linux based Cluster management tool available for the administrator.

2.2 PROPOSED SYSTEM

The system developed will have high computation speed, will be powerful based on the number of nodes provided with cheap off the shelf hardware.

Linux Clusters are developed with the help of dedicated nodes. A network of dedicated nodes is connected to a server. The server will have a parallel computing software. The parallel computing software is used installed in the server.

The software will split the processes and assign the processes to the dedicated nodes. The nodes will perform the necessary computation and will return the result to the server.

Multithreading is supported hence even a single process can be split up and assigned to different nodes in the system hence increasing the computation speed.

Linux clusters are fault tolerant hence even when hardware fails, user will experience only a drop in the performance.

The Linux Cluster management tool is developed for easy maintenance of the clusters. The cluster management tool contains

- Database
- Server Module
- Client Module

2.3 ADVANTAGES OF PROPOSED SYSTEM

The advantages of Linux Clusters are:

1. Linux Clusters are cheap.

2. Linux Clusters are stable.

3. Linux Clusters are easy to upgrade.

4. Linux Clusters are fault tolerant.

The advantages of Linux Cluster Management Tool are:

- The Cluster administrator can easily identify the system characteristics of each cluster node connected in the network from the server module.
- It saves a lot of time.
- The cluster administrator can easily identify the system characteristics, user accounts, details of projects from the database
- Data can be easily transmitted to different nodes easily
- Remote Shutdown is possible.

CHAPTER 3

SYSTEM DESIGN

3.1 DESIGN CONSIDERATIONS

Before building a cluster, the type of problems to be solved must be considered. Different kinds of applications will actually run at different levels of performance on different kinds of clusters. The way we connect the cluster together can have a great impact on its efficiency.

When designing a cluster one of the things we should decide is whether the cluster should be homogeneous or heterogeneous.

HOMOGENEOUS CLUSTERS

In homogeneous clusters, we will be putting together a cluster in which every single node is exactly the same, from the motherboard and the memory, to the disk drives and the network controller cards.

HETEROGENEOUS CLUSTERS

Heterogeneous clusters come in two general forms. The first, and most common, are heterogeneous clusters made from different kinds of computers. For example, a few Sun SPARCstation IPXs, a few Intel 486 machines, and a DEC Alpha. The second common kind of heterogeneous cluster is made from different machines in the same architectural family: for example, a collection of Intel boxes where machines are of different generations.

The cluster developed, a heterogeneous cluster, consists of nodes of both Intel and AMD Family.

3.2 HARDWARE FOR CLUSTERS

When clusters are built, the fundamental aspects to be considered are

- Should we use existing hardware
- Should we build all the hardware from components
- Should we use commercial off the shelf components

Clusters can be built out of almost any kind of hardware. The factors that will probably play into the decision of building a cluster are a trade-off between time (i.e., convenience) and money.

3.3 CLUSTER CONFIGURATION

Once the hardware architecture is decided the next major task to be accomplished is to decide how to connect the individual nodes, to start the transmutation from “piles of boxes” to “parallel Linux supercomputer”.

There are a number of ways to connect a cluster of machines, some of the topologies used are:

- The Network of Workstations (NoW)
- Cubes and Hypercubes
- Meshed Designs
- Rings

3. "**MySQL Reference Manual"** for version 5.0.1-alpha.

4. "**Voodoo's Introduction To Javascript**", Stefan Koch.

Websites:

http://www.linuxhorizon.org

http://www.howstuffworks.com

http://www.echoecho.com

http://www.htmlgoodies.com

http://www.phpfreaks.com

http://www.100webspace.com

http://www.php.net/

http://www.mysql.com

www.csm.ornl.gov/pvm/EuroPVM97/linux PROJECT -MPI

www.csm.ornl.gov/pvm/EuroPVM97/sld007.htm

www-unix.mcs.anl.gov/dbpp/text/node4.html

http://webct.ncsa.uiuc.edu:8900/SCRIPT/MPI/scripts/serve_home - MPI

http://webct.ncsa.uiuc.edu:8900/webct/public/home.pl

http://yara.ecn.purdue.edu/~pplinux/PPHOWTO/pphowto.html#toc3

http://debianlinux.net/system_management.html

The cluster is configured as a network of workstations. In this setup a simple LAN is usually used by individual nodes to form a cluster. The clustering software such as PVM or MPI is used that allows the LAN to be used effectively as a parallel processor.

3.4 BUILDING THE MASTER NODE

The master node is the smartest node in a cluster. It is the system that has all of the software required to run the whole cluster. The master node has a large number of programs and configuration files that will allow you and other users of cluster to treat the master node as if it were the entire cluster itself, for the purpose of running programs.

From a management perspective, the master node will be able to control and monitor all aspects of the operation of the cluster. The addition or removal of compute nodes will also take place through software installed. The master node must be more heavily equipped than the slave or compute nodes.

3.5 MASTER NODE NETWORK CONFIGURATION

A system management utility is run to customize the networking parameters, the utility will run as a background process. The necessary details such as system name, domain name and the IP address are given as input.

3.6 MASTER NODE CLUSTERING SOFTWARE INSTALLATION

The master node must be configured in such a way that it has all the cluster software, some of the software are message passing libraries,

administrative tools such as database for user administration and system administration and Linux monitoring tool.

3.6.1 BUILDING A BOOT SERVER

The master node must be configured to provide information to slave/compute nodes and automatically initialize them.

Building a boot server involves configuring several important subsystems, including the Domain Name Server and the Dynamic Host Configuration Protocol configuration file.

The Dynamic Host Configuration Protocol (DHCP) allows an administrator to set up a mechanism for configuring client computers at boot time, without having to keep configuration information on a local disk drive.

3.6.2 MESSAGE PASSING INTERFACE (MPI)

Parallel processing is a form of computing in which a number of activities are carried out concurrently so that the effective time required to solve the problem is reduced. In the previous days, parallel processing was used for such thing as large scale simulations (e.g. molecular simulations, simulation of the explosion of an atomic bomb etc), solving large number crunching and data processing problems (e.g. compiling the census data) etc. However, as the cost of hardware is decreasing rapidly, parallel processing is being used more and more in routine tasks. Multiple processor servers have been in existence for a long time. Parallel processing is also used in our own PC too. For example, a graphics processor working along with the main processor to render graphics on your monitor is also a form of parallel processing.

However, apart from the hardware facilities for parallel processing, some software support too is required so that we can run the programs in parallel and coordinate their execution. Such coordination is necessary due to the dependencies of the parallel programs on one other. This will become clearer when we work through an example. The most widely used method to achieve such coordination is *message passing* in which the programs coordinate their execution and in general communicate with each other by passing *message's* to one other.

Message Passing

Message passing is a model for interactions between processors within a parallel system. In general, a message is constructed by software on one processor and is sent through an interconnection network to another processor, which then must accept and act upon the message contents. Although the overhead in handling each message (latency) may be high, there are typically few restrictions on how much information each message may contain. Thus, message passing can yield high bandwidth making it a very effective way to transmit a large block of data from one processor to another. However, to minimize the need for expensive message passing operations, data structures within a parallel program must be spread across the processors so that most data referenced by each processor is in its local memory.

MPI is a library specification for message-passing, proposed as a standard by a broadly based committee of vendors, implementors, and users. The MPI standard is available. MPI was designed for high performance on both massively parallel machines and on workstation clusters. MPI is widely available, with both free available and vendor-supplied implementations. A number of MPI home pages are available. MPI was developed by a broadly based committee of vendors, implementors, and users.

3.7 BUILDING THE SLAVE/COMPUTE NODE

Since the cluster developed is a Network of Workstations (NOW) cluster, the network interface cards on each node in the cluster, including the master node, are connected to a network hub or switch.

The slave node spends all of its time in user mode, executing user jobs. The slave node configuration involves installing the Linux operating system and configuration of the network.

3.8 DATABASE DESIGN

3.8.1 ER-MODEL

Linux Cluster Management tool needs more than one table to store the user details,cluster details and the node details.

The figure3.1 is the Entity-Relationship Diagram used for Linux Cluster Management.

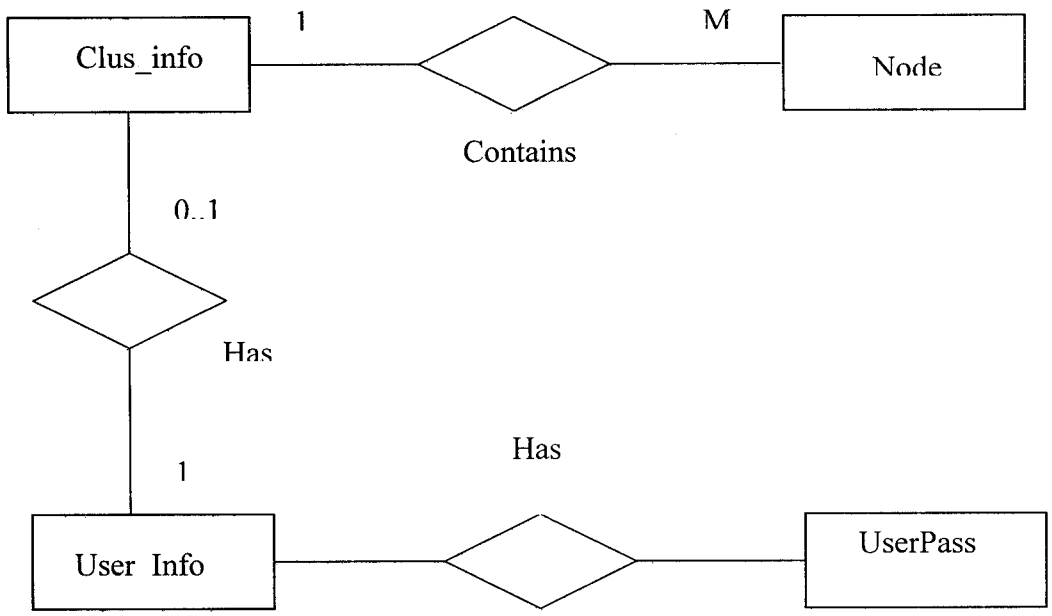


fig 3.1:Entity Relationship Diagram

3.9 TABLE DESIGN

There are 4 tables involved in the system.

FIELD NAME	TYPE
Cluster_id	Varchar(8)
Cluster_Name	Varchar(12)
Master_Node	Varchar(1)

Table 3.1: Clus_info

FIELD NAME	TYPE
Node_id	Varchar(12)
Cluster_id	Varchar(8)
mac_address	Varchar(18)
Ip_address	Varchar(16)
Hardware_type	Varchar(12)
Processor_speed	Integer
Memory	Integer
Total_disk_space	Integer
Master	Varchar(1)

Table 3.2: Node

FIELD NAME	TYPE
User_id	Varchar(12)
Password	Varchar(12)

Table 3.3: User _Pass

FIELD NAME	TYPE
user_id	Varchar(12)
Password	Varchar(12)
First_Name	Varchar(12)
Last_Name	Varchar(12)
Email	Varchar(30)
Phone_no	Varchar(12)
Status	Varchar(10)
Creation_date	Date

Table 3.4: User_Info

3.10 INPUT / OUTPUT DESIGN

The Input to Linux Cluster Management System is the information collected from the slave nodes based on the requirements of the cluster administrator from the master. The collected information is stored as a file. All these are background process. The output screens will be displayed only in the master node. The database consists of information such as node information, user information.

Main Screen

The administrator will log in into the Linux Cluster Management System by starting the application. The Screen consists of all the functionalities and the names and IP addresses of all the slave nodes online.

File Transfer

The file transfer screen is used to transfer a file from client machine to the server machine or vice versa. The source file can be selected using the file picker box and the destination path can also be selected from the list displayed. By clicking the respective button the files can be transferred easily.

Remote Screen Capture

The Remote Screen capture screen will display the desktop of the requested node and the image captured is refreshed for particular period so that any change in the desktop of the cluster node will be immediately noticed from the Master node.

Remote Shutdown

The Remote Shutdown will display the names and IP addresses of all the slave nodes, the administrator must select the node and press the shutdown button.

Memory Usage

The memory usage screen will display the attributes of both the primary and the secondary storage devices of the slave node requested.

Bandwidth Calculation

The bandwidth calculation screen displays the information about the bandwidth used for the transmission of data.

Automatic Management

The automatic management module will run as a background process in all the slave nodes .It will monitor the CPU temperature and the primary memory of the system.

Remote Installation

The remote installation module is used to install the Linux operating system and necessary software in the slave nodes from the master node. The administrator need not go to each slave node for installation.

Adding Nodes

The add node screen is used to add new nodes to the cluster .Some information to be fed as input are node name, Mac address, ip address etc.

Add Cluster

The add cluster screen is used to add a new cluster.

Add User

The add user screen is used to add new users for the cluster.

Show Users

The show users screen is used to show all the registered users.

Available Clusters

The available clusters screen is used to display all the clusters currently available.

3.11 ARCHITECTURAL DESIGN

Architectural Design is concerned with refining the conceptual design of the system, identifying internal processing functions, decomposing high level functions into sub functions, defining internal data streams and data stores and establishing relationships and interconnections among functions, data streams and data stores.

Linux Cluster Management Tool – Module Overview

The Linux Cluster Management Tool consists of following modules

- File Transfer Module
- Remote Screen Capture Module
- Remote Shutdown Module
- Memory Usage Module
- Bandwidth Calculation Module
- Automatic System Management Module

- Remote Installation module
- Node Administration
- Users Administration

File Transfer Module

The File Transfer Module is used to transfer a file form one machine to another machine.

There are two types of file transfer

Cluster Node to Master Node

Master Node to Cluster Node

The source file is selected in the source machine and the file is compressed and the compressed file is written onto the socket. The destination machine reads the file, decompresses the file and copies the file to the destination path.

Remote Screen Capture Module

The Remote Screen capture module captures the desktop of the requested node and then it transfers the image to the server .The image captured is refreshed for particular period so that any change in the desktop of the cluster node will be immediately noticed from the Master node.

Remote Shutdown Module

The Remote Shutdown module is developed to shutdown the slave nodes from the Master node when the need arises. Once the slave node to be shutdown is selected, the remote shutdown module will send a message to the slave node to shutdown the system.

Memory Usage Module

The memory usage module is used to display all attributes of the primary and secondary storage devices present in the client machine. The client program checks the primary and the entire secondary storage device and writes the details to a file. After completing the process the file is written to the socket. The server machine reads the file and displays the information to the administrator.

Bandwidth Calculation Module

The bandwidth calculation module displays the information about the bandwidth used for the transmission of data.

Automatic Management Module

The automatic management module will run as a background process in all the slave nodes .The module will monitor the CPU temperature and the primary memory of the system, when both the values reach particular threshold value automated information will be sent to the master node.

Remote Installation

The remote installation module is used to install the Linux operating system and necessary software in the slave nodes from the master node. The administrator need not go to each slave node for installation.

Node Module

The node management module will keep track of all of the nodes in the cluster. We can define sub clusters of machines.

User Module

The user management module is used to add, delete, activate or suspend access to users of cluster.

3.12 PROCESS DESIGN

System Process Architecture

The following figure fig (3.2) represents the overall system architecture of the Linux Cluster Management System.

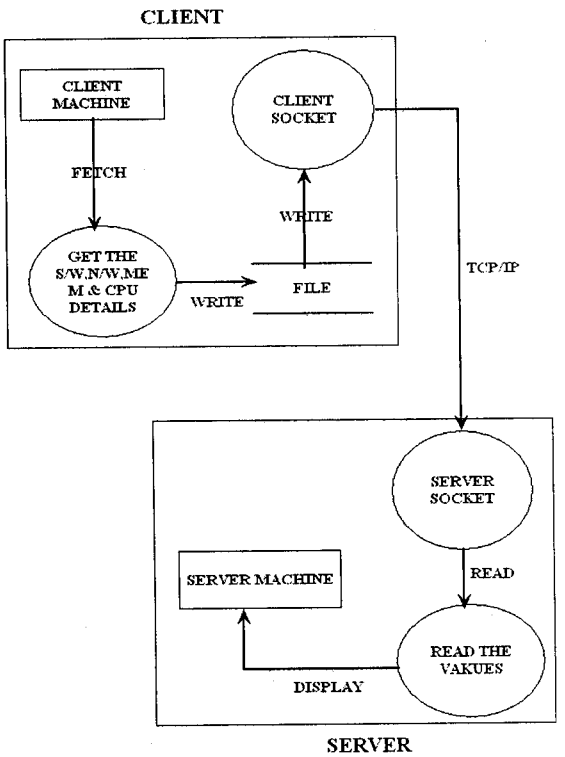


Fig 3.2 : Architecture of System

CHAPTER 4

SYSTEM TESTING AND IMPLEMENTATION

The system testing deals with the process of testing the system as a whole. This is done after the integration process. Moving through each module from top to bottom tests the entire system. The verification and validation processes are then carried out. The errors that occur at testing phase are eliminated and a well functioning system is developed.

Test case design focuses on a set of techniques, which meets all testing objectives, which are mentioned below.

1. Testing is a process of executing a program with the intent of finding an error.
2. A successful test is one that uncovers an as yet undiscovered error.

Testing demonstrates that software functions work according to specifications. In addition data collected from testing provides a good indication of software reliability and some indication of software quality as a whole.

Testing results are used for detecting errors. Critical modules are tested as early as possible. The following tests are carried out.

4.1 TESTING METHODS

Unit Testing

It focuses verification efforts on the smallest unit of software design, the module. This is also known as **Module Testing.** The modules are tested separately. This testing is carried out during programming stage itself.

Validation Testing

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the users.

After validation test has been conducted one of the two possible conditi ons exist

1. The function or the performance characteristics confirm to specification and are accepted
2. A derivation from specification is uncovered and a deficiency list is created.

Output Testing

After performing the validation testing the next step is output testing of the proposed system since no system is useful if it does not produce the required output in the specific format. Asking the users about the formats required by them tests the outputs generated or displayed by the system under consideration.

User Acceptance Testing

User acceptance of a system is a key factor for the success of any system. The system under consideration is tested for user acceptance by

constantly keeping in touch with prospective system users at the time development and making changes whenever required.

TEST CASES

Some of the test cases used for testing the system and the descriptions of those test cases are as follows,

S.No	Test Case Description	Expected Result	Status	Verification Method
1	Sessions should expire after logout	The previously visited pages should not be displayed	Pass	Directly type any page address
2	Sessions should expire after the browser window gets closed	The previously visited pages should not be displayed	Pass	Open the browser again and open the history pages
3	The Administrators should only be allowed to login into this system	The system should prompt for the user about his membership if not the Administrator	Pass	Login Using different ID's and check the interface
4	The User should be authenticated properly	The Users should not be allowed to login with wrong passwords	Pass	Login using different combinations of user ID's and Passwords

Table 4.1 Test Cases

4.2 IMPLEMENTATION

The implementation phase of software development is concerned with translating design specification into source code. The primary goal of implementation is to raise source code and internal documentation so that conformance of the code to its specification can be easily verified, and so that debugging, testing and modification are eased. This goal can be achieved by making the source code as clear and straightforward as possible. Simplicity, clarity and elegance are the hallmarks of good programs; obscurity, cleverness and complexity are indication inadequate design and misdirected thinking.

Source code clarity is enhanced by structured coding techniques, by good coding style, by appropriate supporting documents, by good internal comments and by the features provided in modern programming languages.

The goal structured coding is to liberalize control flow through a computed program so that the execution sequence follows the sequence in which the code is written. The dynamic structure of a program as it executes then resembles the static structure of the written text. This enhances readability of code, which eases understanding, debugging, testing, documentation and modification of programs. It also facilitates formal verification of programs. The structure coding techniques are as follows:

- Single entry, Single exit constructs
- Efficiency considerations
- Data Encapsulation
- Recursion

IMPLEMENTATION ON LINUX

The system is implemented in Red Hat Linux 9.0 with the web server being Apache v2.0 and the client side browser being the Mozilla Fire fox v1.0. The

MySQL Version used is 4.1.6 and the PHP Engine version embedded into the apache server is 4.3.9.The java version is 1.5

4.3 MAINTENANCE

Maintenance is the enigma of system development. It holds the software industry captive typing up programming resources. It could be described as the symmetric process of changing the software that is already in operation in order to prevent system failures and to improve the performance. Software maintenance involves keeping software interfaces simple and standard, paying particular attention to troublesome modules, replacing faulty components and generally planning to replace components that are ole, obsolete, faulty, or at risk for imminent failure.

There are several factors that require to be maintained. They are

- Hardware platforms change or become obsolete.
- Operating system change.
- Compiler change
- Language standard's change.
- Communication standard's change
- Graphical user interface change.
- Related application software package change.
- New system being added to the network.

Maintenance can be classified into

- Adaptive maintenance
- Perceptive maintenance
- Preventive maintenance
- Corrective maintenance

Adaptive Maintenance

It deals with adapting software change in the environment. It does not lead to changes in the system functionality.

Perceptive Maintenance

It mainly deals with accommodating new or changed users requirements. It also includes activities to increase the system performance or to enhance its user interface. The objective of perceptive maintenance should be to prevent failures and optimize the software.

Preventive Maintenance

Preventive maintenance concerns activities aimed at increasing the system's maintainability such as updating documentation adding comments, improving modular structure of the system.

Corrective Maintenance

This deals with the repair of faults found. Some of the major causes of maintenance problems are Unstructured Code and Maintenance programmers have insufficient knowledge of the system and on application domain.

CHAPTER 5

CONCLUSION

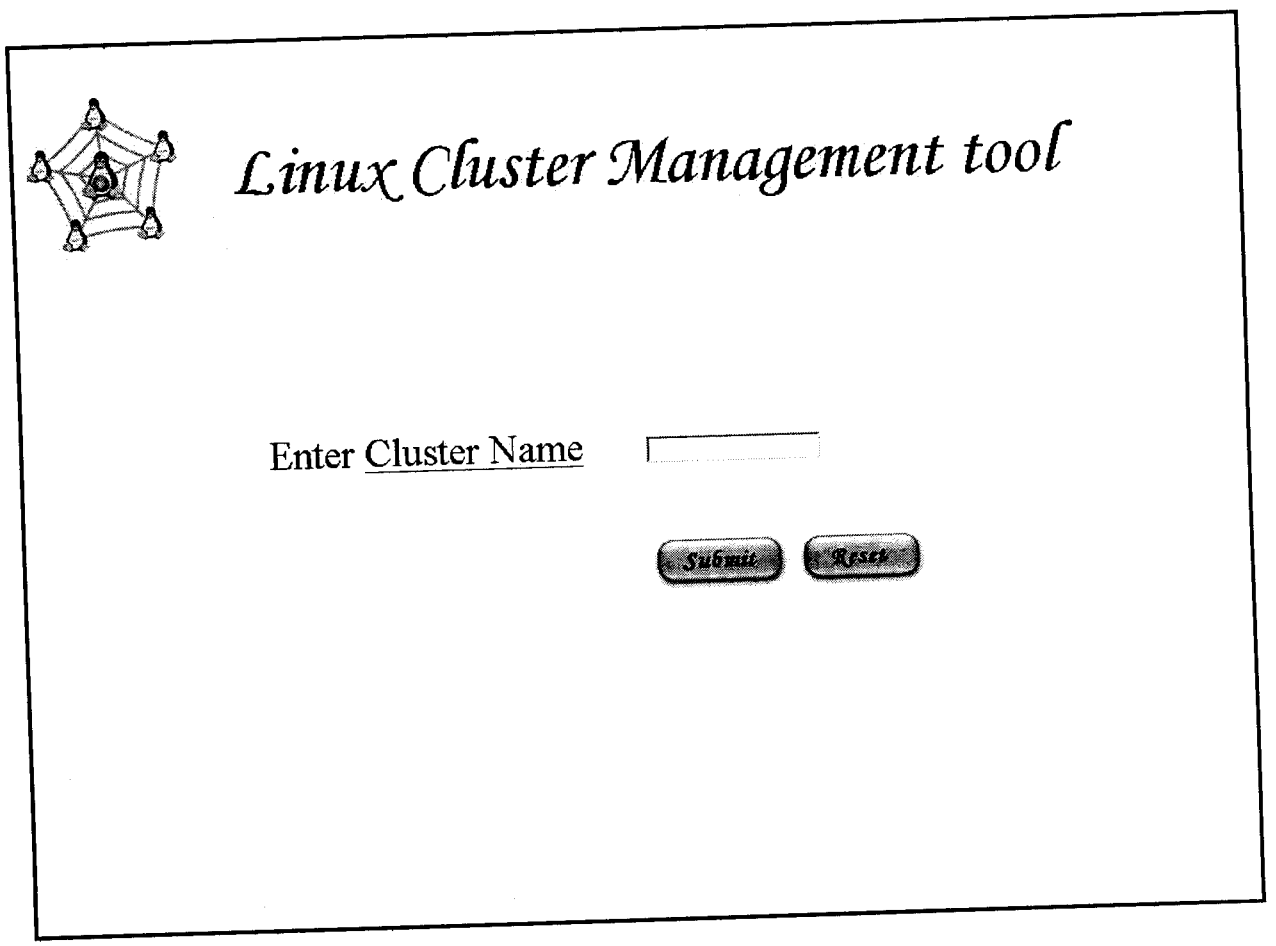
With the development of Linux Clusters the high performance computing has been made possible with off the shore components.

With the Linux Cluster Management tool the administration of cluster have been made easier by maintaining a database about the users and the clusters available. The remote shutdown and remote installation module allows the administrator to shutdown and install software for entire cluster from the master node. The automatic management module will shut down the slave nodes automatically when the specific threshold value is reached. The file transfer module will provide an easy way to transfer files from master node to the slave nodes. Thus this Linux Cluster Management tool provides a complete solution for an administrator in managing the cluster.

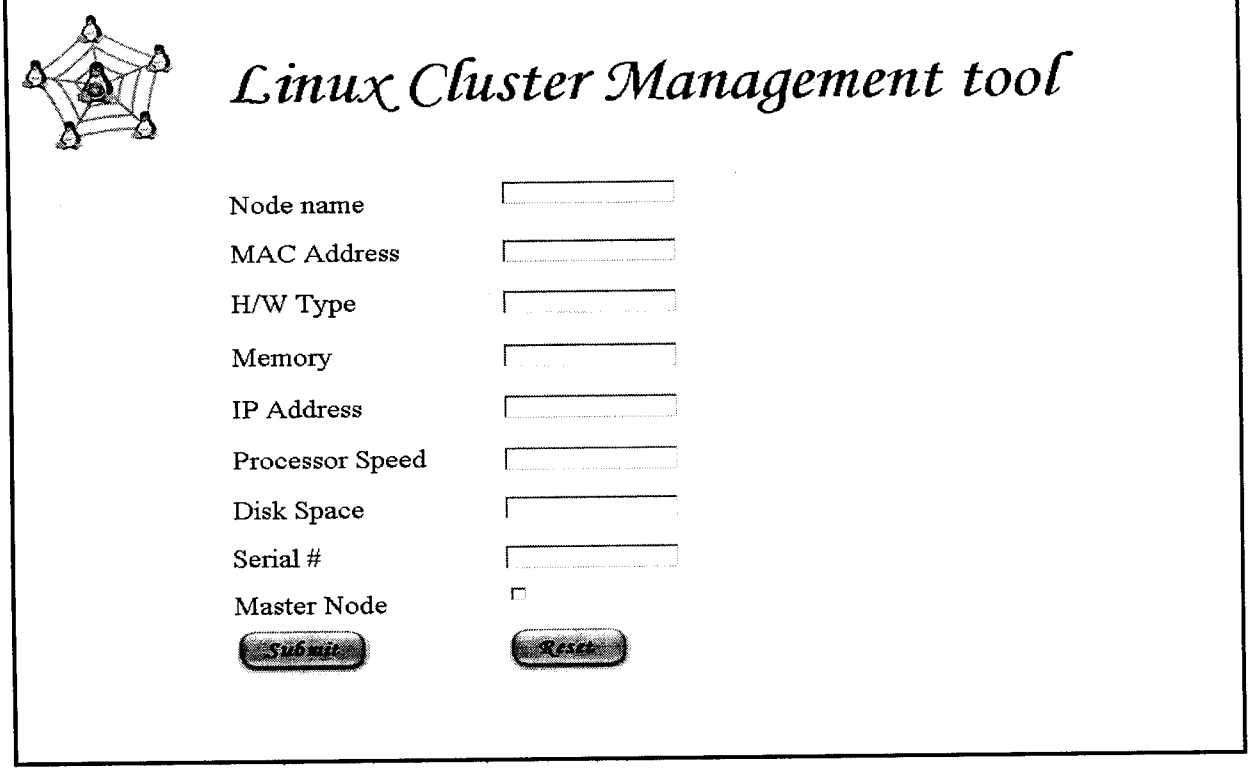
APPENDICES

SCREEN SHOTS

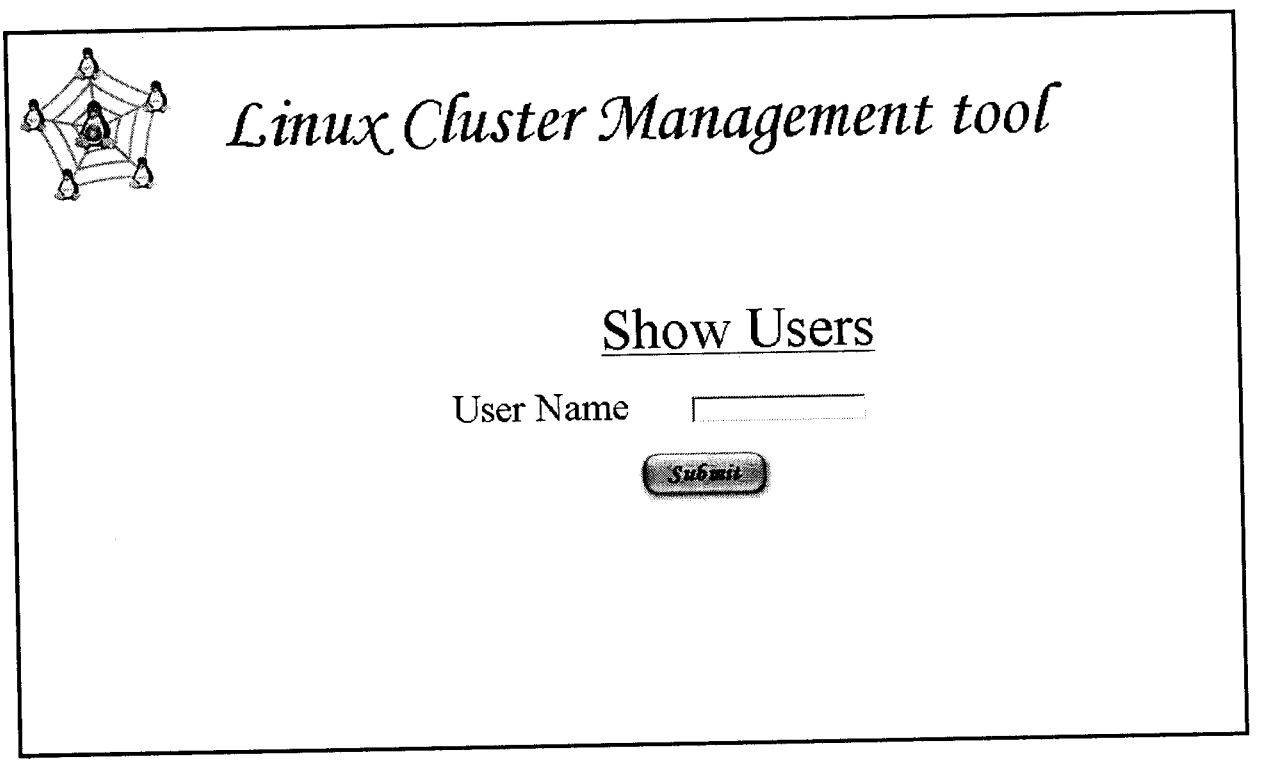
ADD CLUSTERS SCREEN



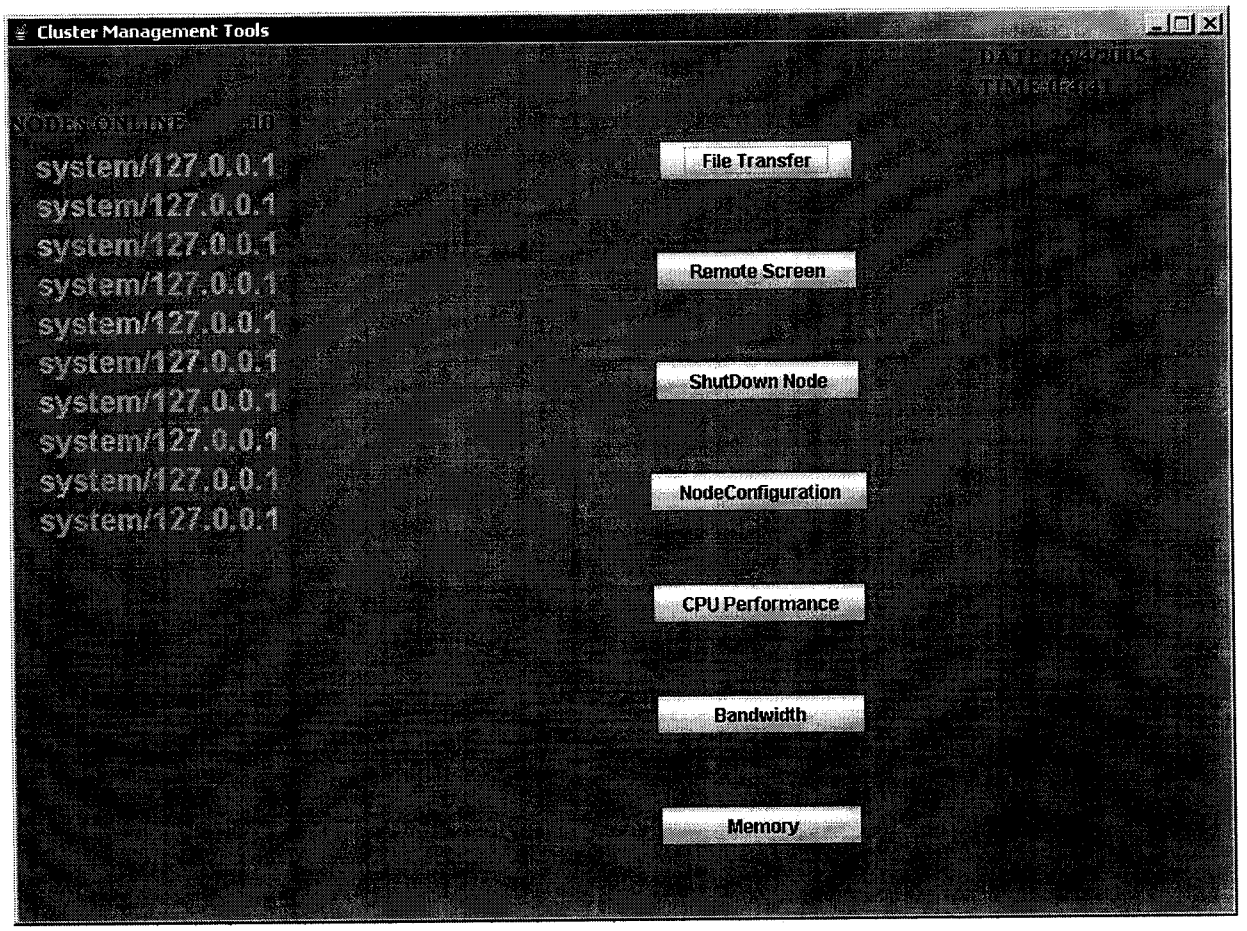
ADD USER SCREEN



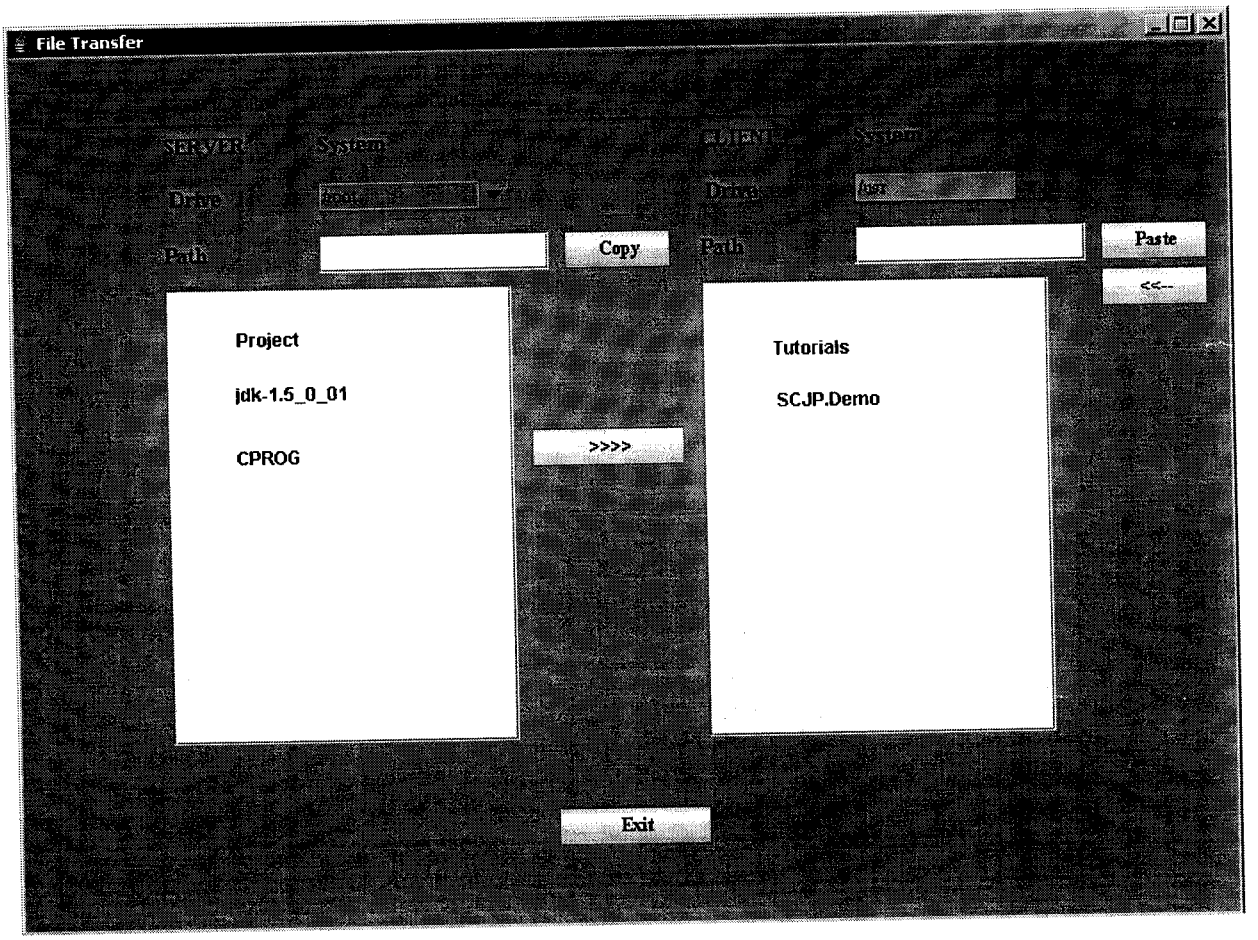
Show Users Screen



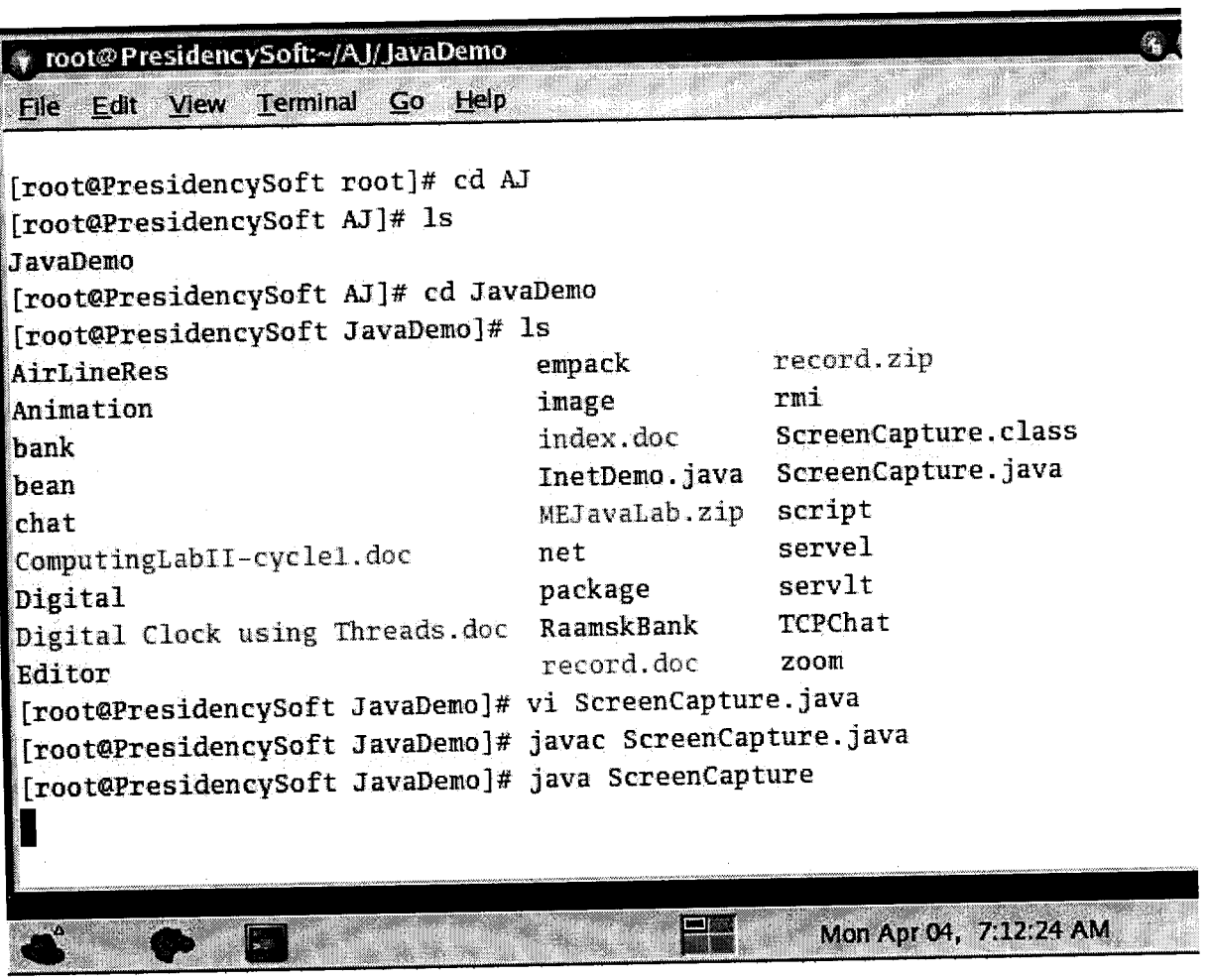
MAIN SCREEN


Cluster Management Tools
NODES ONLINE
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
system/127.0.0.1
File Transfer
Remote Screen
ShutDown Node
NodeConfiguration
CPU Performance
Bandwidth
Memory

FILE TRANSFER SCREEN



REMOTE SCREEN CAPTURE



REFERENCES

Text Books:

1. David HM SPECTOR, "**BUILDING LINUX CLUSTERS",** O'Reilly.
2. Vikram Vaswani, "**MySQL-The Complete Reference**", Tata McGraw Hill, 2004.
3. Nigel McFarlane, Andrea Chiarelli, James De Carli, Sing Li, Stuart Updegrave, Paul Wilton, "**Professional JavaScript**", Wrox Press Ltd., 2000.
4. George Schlossnagle, "**Advanced PHP Programming**", Pearson Education, 2004.

E-Books:

1. "**PHP Manual**", Stig Sæther Bakken, Alexander Aulbach, Egon Schmid, Jim Winstead, Lars Torben Wilson, Rasmus Lerdorf, Andrei Zmievski, Jouni Ahto
2. "**Professional PHP Programming**", Jesus Castagnetto, Harish Rawat, Sascha Schumann, Chris Scollo, Deepak Veliath