UNIVERSITI UTARA MALAYSIA

SERVER-BASED OPEN SYSTEM IMPLEMENTATION FOR ICOSYS (ICOSYS2)

(S/O CODE: 11072)

SUHAIDI HASSAN PhD (Team Leader)
AHMAD HANIS SHABLI
AMRAN AHMAD
FAZLI AZZALI
NOOR IZZAH YAHYA

FACULTY OF INFORMATION TECHNOLOGY
FEBRUARY 2005

Abstract

Information and communication services are developed to enhance the utilization of the ICT infrastructure. These services can be implemented either using proprietary or open source software platforms. This work focuses on the implementation of the open source services at the Faculty of Information Technology (FTM), Universiti Utara Malaysia. The study is part of the ICOSYS research intitiative. The underlying vision of ICOSYS2 is to create a conducive and efficient working environment using open source technologies. ICOSYS2 consists of four basic services that comprises file storage, application, printing and backup services. Firstly, ICOSYS2 files and storage services provide users with the ability of centrally storing and sharing data. Secondly, the ICOSYS2 application services provide access to the world of Linux-based open source applications. Thirdly, ICOSYS2 printing services allow users to send their print jobs to respective printers across the FTM network and pave the way for centralized printing administration. Finally, the ICOSYS2 backup services outlines a tertiary mean to backup the users' data. It is hoped the implementation of this ICOSYS2 project can facilitate academic and administrative tasks at FTM.

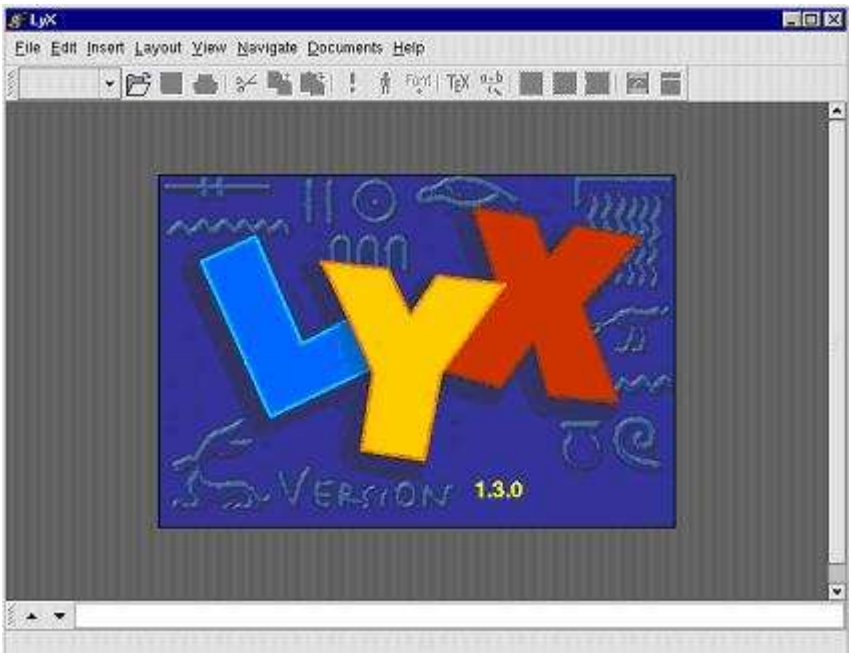
Acknowledgement

"Uncle Cosmo, why do they call this a word processor?"

"It's simple, Skyler. You've seen what food processors do to food, right?"

— Jeff MacNelly in "Shoe"

Our *extraordinary* tribute goes to **LyX**[1], the increasingly popular open-source document processor. With LyX, the process of writing this report is made simpler. This whole document is prepared using LyX. LyX document processor has been made available on our ICOSYS2 server. After all, this is our way to say that - *we do practice what we preach in this report!*



[1] **LyX is what?** LyX is the first WYSIWYM (What You See Is What You *Mean*) document processor. It is an advanced open source document processor that encourages an approach to writing based on the structure of your documents, not their appearance. LyX lets you concentrate on writing, leaving details of visual layout to the software. LyX produces high quality, professional output – using LaTeX, an industrial strength typesetting engine, in the background; LyX is far more than a front-end to LaTeX, however. No knowledge of LaTeX is necessary to use LyX, although it will give a user more power. LyX is stable and fully featured. It has been used for documents as large as a thesis, or as small as a business letter. Despite its simple GUI interface (available in many languages), it supports tables, figures, and hyperlinked cross-references, and has a best-of-breed math editor.

More? Surf to **http://www.lyx.org**

Contents

List of Figures

List of Tables

Abbreviations

AFS	Andrew File System
CUPS	Common UNIX Printing Service
FTM	Faculty of Information Technology, Universiti Utara Malaysia
GNU	Gnu is Not Unix, an on-going computing project of the Free Software Foundation
GPL	GNU Public Licence
GUI	Graphical User Interface
ICOSYS	Information and Communication System Services
IPP	Internet Printing Protocol
KDE	K Desktop Environment
LAN	Local Area Network
LPD	Line Printing Daemon
Mac	Apple Inc.'s Macintosh
NetBIOS	Network Basic Input Output System
NFS	Network File System
OS	Operating System
OSS	Open Source Software
PC	Personal Computer
RPM	Redhat Package Manager
SAN	Storage Area Network
SCSI	Small Computer System Interface
SMB	Server Message Block
SSH	Secure Shell
URL	Uniform Resource Locater

UUM	Universiti Utara Malaysia
VFS	Virtual File System
Windows	Microsoft Windows(R), also Windows (9x/2000/Me/XP), the registered trademarks and copyrights of the Microsoft Corp.

Chapter 1

Introduction

Faculty of Information Technology, Universiti Utara Malaysia (FTM) is currently well-equipped with the ICT infrastructure. All of its lecturers and administrative staff are provided with workstations, which are connected to the 100Mbps switched Ethernet local area network (LAN). The FTM network is linked to the University's Gigabit-Ethernet backbone.

Although FTM ICT infrastructure looks sufficiently impressive, FTM is indeed lacking a 'system' environment that provides the required system services which add value to this hi-tech infrastructure. Such system services includes file storage services, application services, printing management and backup services. For example, printing and backup tasks are managed individually by users instead of by the systems, which certainly can affect users' productivity. These services are not uncommon in advanced IT organizations around the world, but not in FTM.

The FTM workstations are of stand-alone nature, relying on a single, Microsoft-based platform environment. Most of these machines run Microsoft Windows-based systems (Win9x/2000/XP). Users works (mainly academic and administrative) are kept in these workstations. There is no centralized file management system, no backup services provided, no power management system, no authentication and security services,

no printing management, and, worst of all, no system policy. Such deficiency can pose serious implications, the most important of which is security. Without file management and authentication systems, FTM workstations are widely exposed to security intrusions. In addition, the risk of data loss is extremely high due to the absence of a centralized backup and recovery service.

All these deficiencies could lead to a less efficient working environment. For example, users have to carry out their own backup routine (if they bother), taking their own measures to protect their system, manage their own file system, recover their lost data etc. All these chores consume users' time and organizational resources. This inefficient working environment in turn could degrade both users' morale and organizational productivity.

Additionally, FTM academic staff is putting too much reliance on the single Microsoft-based platform environment. Strategically, such reliance could affect FTM's competitiveness in the future, as Malaysia is moving towards adopting an open source system environment. For this reason, FTM must assume proactive steps in adopting an open source platform in order to become a key player at the forefront in implementing open source systems; otherwise FTM will be left behind in the industry. Hence, FTM must initiate an effort to introduce another system platform, that is a Linux-based open source platform, that can be used to support academic, and possibly administrative activities in FTM.

In short, FTM has a relatively good networking infrastructure, but lack the basic services that are commonly needed to enhance productivity in a networked environment. Thus, providing such services is a must for the FTM communities. The implementation of these services can add value to utilization of the networked environment. In this study, we implement several basic system services in FTM using the open source approach.

1.1 Research Motivation

The motivation for addressing these concerns is extremely clear. As a major information and communication technology (ICT) organization with the prestigious MSC status, FTM must not bear the fact that it is lacking a conducive computing system environment that, if known to the outside world, could tarnish its image as a reputable ICT organization in Malaysia.

While the FTM lecturers are preaching their students on how to manage and administer ICT environments, the lecturers themselves are not familiar or exposed to such a conducive and practical ICT environment. Hence, FTM must take proactive steps to create a stable, state-of-the-industry computing environment that the University can be proud of.

ICOSYS2 is a research initiative and effort geared toward enabling an efficient utilization of the available ICT infrastructure at the Faculty of Information Technology. The main goal of ICOSYS2 is to create and implement a stable, state-of-the-industry system environment that delivers the integrated, multi-platform open system services for supporting the main academic activities of the Faculty, namely teaching, research and consultancy.

1.2 Research Context

This study is a part of the Information and Communication System Services (ICOSYS) project. The project attempts to enable an efficient workgroup computing environment within FTM by implementing basic centralized system services such as file, printing and backup services. The FTM ICOSYS intitiative comprises four main research exercises namely;

- Proprietary System Implementation for ICOSYS (ICOSYS1). This involves research on proprietary system (Microsoft-based) user and system management

issues (such as system implementation, backup and retrieval and file management).

- Open System Implementation for ICOSYS (ICOSYS2). This involves research on open system (Linux-based) user and system management issues (such as system implementation, backup and retrieval and file management).

- Managing Interoperability issues of ICOSYS (ICOSYS3). This involves research on managing a heterogenous multi-platform system environment that includes issues like multi-platform authentication services and cross platform interopability issues.

- Security consideration for ICOSYS (ICOSYS4). This research involves studying the security impact on ICOSYS. This includes studying vulnerability of the ICOSYS in facing possible security threats as well as methods to prevent such attack.

For the ICOSYS2 implementation, we have selected four basic system services namely file storage, application, printing management and backup services. The file storage service enables users to centrally store their files. With the centralized file storage system, user documents and files can be effeciently shared and periodically backed up by the system.

With the application services, users are able to use applications that are not available on their workstation. For example, if a Microsoft Windows-based user needs to use an application that only exists in the Linux-based platform, the user will just need to login to the server and run the needed application from their Microsoft Windows-based system. With the ICOSYS application service, it is possible to reduce the need for maintaining every application on every workstation.

The ICOSYS printing services provide a centralized system-wide printing management that manages printing queues, printing quotas, and other printing tasks. These

printing management services are important to enhance current printing capabilities among FTM users and to manage printing resources more effectively.

The ICOSYS2 backup services bring backup facility into existence in FTM. These services are intended to boost staff productivity, save time and energy to recover data losses, and to enhance the workgroup collaborative environment within FTM. These features are important for users so that they can recover their files in any unfortunate events, such as hard disk failure. Users can restore their work without much trouble.

1.3 Methodology

For the purpose of the study, we have adopted a methodology for implementing the service using the Windows 2000 test lab strategy from Microsoft. The methodology [3] is well constructed for Windows-based implementation for test lab. We redefine the methodology for our open source system implementation.

The methodology is divided into three phases; strategy, design and build phases. The methodology was adopted with several adjustments to suit our needs for an open system implementation. Figure 1.1 shows the three phases and the detail taken in performing the methodology.

- **Phase #1**: **Strategy phase**

  The initial phase of our implementation was to determine our goal and srategy in implementing such services. The general strategy for ICOSYS2 was describe in the proposal for fund approval.

- **Phase #2: Design phase**

  In the design phase, design issues for service implementation, such as logical and physical network design, and hardware and software requirements are determined. How the services will be deployed and what strategy to deploy these

services are also established.

- **Phase #3: Build phase**

  This phase is the actual implementation of the selected services in the network, which includes configuration and setup and running of the services that are ready to be tested and used.

The implementation of the methodology is described further in Section 2.6.

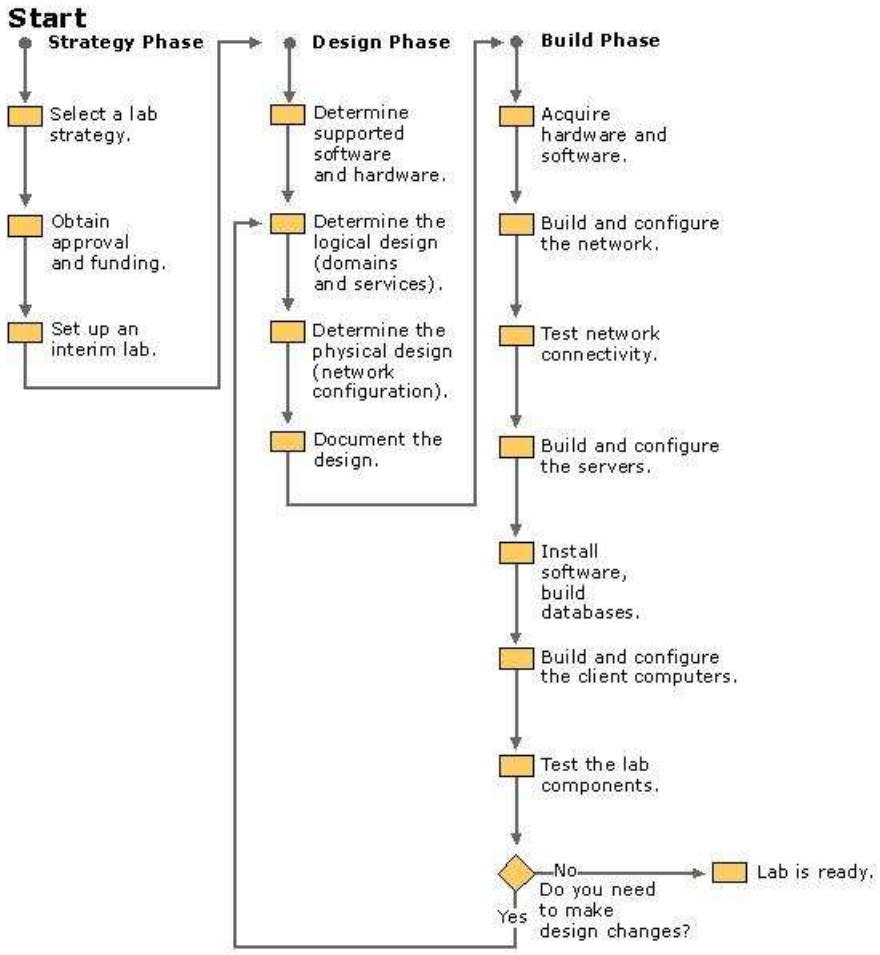


Figure 1.1: Windows 2000 Test Lab [3]

1.4 Aims and Objectives

The study attempts to implement an integrated information and communication systems services using an open source system in FTM. The open source system implementation involves research on the open source system (Linux-based) user and system requirements, as well as user and system management issues (such as system implementation, backup and retrieval, and file management services).

The aim of this research is to implement the basic network services in FTM that include file, application, printing and backup services mainly by using the open source-based system.

The objectives of this research can be summarized as follows:

- To design an integrated information and communication systems services using an open source system on the FTM network.
- To implement the integrated information and communication systems services, namely file, application, printing and backup services using an open source system on the FTM network.

1.5 Report Overview

This report presents our work on the design of an integrated information and communication systems services using an open source system, and its implementation on the FTM network. The rest of the report is organized as follows:

Chapter 2 outlines the technological and industrial outlook and trends as well as our strategy for implementing ICOSYS2 services. In this chapter, we provide the research background from the perspective of the state-of-the-industry and the state-of-the-practice. The practice of the current technology and success stories of several well known industries regarding the implementation is also reviewed.

Chapter 3 describes our detailed implementation of the network file storage services, which include a detailed description on the service implementation and configuration issues from the system administrator and user perspectives.

Chapter 4 details the implementation of the ICOSYS2 application services where the implementation on X-server will be discussed. The configuration on how to turn on the services is briefly described.

Chapter 5 discusses our work on the implementation of ICOSYS2 network printing services. The service is used to manage all the printer resources available at FTM under one single management scheme.

Chapter 6 describes our implementation of the ICOSYS2 backup services to provide tertiary backup solution for user data. The strategy regarding the implemenation are also discussed.

Finally, in Chapter 7, several contributions, future works, and a conclusion to our report have been put altogether.

Chapter 2

Background

2.1 Introduction

In this chapter, we provide the background of this research. This includes outlining the technological and industrial outlook and trends as well as our strategy for implementing ICOSYS2 services. Section 2.2 through to Section 2.5 review the state-of-the-practice of implementing open-source system services. Within these sections, we also provide various surveys on the industrial perspectives of the implementation of these services. Section 2.6 outlines our ICOSYS2 implementation strategy. Section 2.7 concludes this chapter.

2.2 File Storage Services

File storage services basically provide users with the ability of centrally storing and sharing data. A distributed file system stores files on one or more computers, called servers, and makes them accessible to other computers called clients, where they appear as normal files. There are several advantages of using file servers. The files are more widely available since many computers can access the servers, and sharing the files from a single location is easier than distributing copies of files to individual clients.

Backup and safety of the information are easier to arrange since only the servers need to be backed up.

The servers can provide large storage space, which might be costly or impractical to supply to every client. The usefulness of a distributed file system becomes clear when considering a group of FTM staff sharing documents. However, more is possible. For example, sharing application software (as discussed in later sections) is an equally good feature. In both cases, system administration becomes easier.

File storage is one of the common network services available in computer networked environments. It enables all computers attached to the network to access files stored on other computers in the same way that they access files on their own disks. Network file storage provides users with a centralized file storage over the network. The advantages of using a centralized file storage are;

- Files on the network server are backed up on a regular basis. If user inadvertently deletes an important file, it can be retrieved from the backup devices. In addition, if users have a hard drive failure, the files the user saved on the network server will not be affected.

- Files can be shared with others. Several people may be working on a document. Storing the file in a shared directory allows each person to directly work on the document as needed.

- If users use several PCs in theie course of work, files on the server are available from any local PC. So, users have access to the files whether he/she is at his/her personal office PC or using a co-worker's PC.

Network storage file services on Linux allow hosts to mount partitions on a remote system and use them as though they are local file systems. This also allows the system administrator to store resources (files or application) in a central location on the network, thus providing authorized users continuous access to them. Many universities

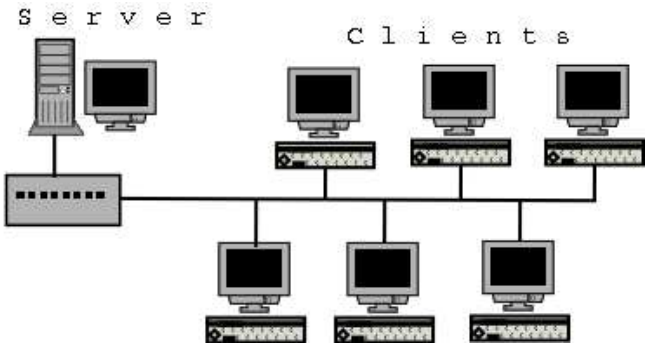


Figure 2.1: Clients Connected to Linux Server

and organizations have already implemented such services and have become the standard campus information technology services. Therefore, these services are become the required components for ICOSYS2 implementation. Providing such services in FTM is very important to the user because it provides means for backup of the user's data and peace of mind if any problem arises at their local workstation (such as hard disk failure or other access problems). Figure 2.1 shows a diagram of several clients connected to a Linux server as an implementation of the network file storage services.

A file server enables all files to be stored in a central location, which accommodates centralized backup strategies (one place to backup all files) and security implementation (depending on the operating system, individuals can be assigned different access rights to stored information). On Linux, two powerful applications, namely NFS and SAMBA, allow network users to access its shared directories from other computers of different operating systems (OS). These applications become the interpreter or the translator for different OSes so that they can communicate with each other. NFS enables Linux clients to communicate with the Linux server while SAMBA enables Windows-based clients to understand (hence, access) Linux file systems.

2.2.1 Network File System (NFS)

NFS [9] is a client/server application designed by Sun Microsystems that allows all network users to access shared files stored on computers of different types. NFS provides access to shared files through an interface called the Virtual File System (VFS) that runs on top of TCP/IP [4]. Users can manipulate and share files as if they were stored locally on the user's own hard disk.

With NFS, computers connected to a network operate as clients while accessing remote files, and as servers while providing remote users access to local shared files. The NFS standards are publicly available and widely used. The downside of NFS is that it is notoriously slow because the process of "teaching" Windows clients how to talk to the UNIX file system. NFS has to "teach" all the clients connected to it to understand UNIX and this results in slow, expensive and unreliable services. NFS works like a charm when communicating between UNIX machines.

2.2.2 SAMBA

SAMBA[1] is an open source/free software suite that provides seamless file and printing services to its clients. It is a suite of UNIX applications that speak the Server Message Block (SMB) protocol[2] and is freely available under the GNU General Public License [14]. Microsoft Windows and the OS/2 operating systems use SMB protocol to perform client-server networking for file and printer sharing and the associated operations.

By supporting this protocol, SAMBA enables computers running UNIX to get into action, communicating with the same networking protocol as Microsoft Windows and appearing as another Windows system on the network from the perspective of a Win-

[1] http://www.SAMBA.org

[2] The Server Message Block Protocol (SMB protocol) provides a method for client applications in a computer to read and write to files on and to request services from server programs in a computer network.

dows client.

SAMBA was originally developed for UNIX but can now run on Linux, FreeBSD and other UNIX variants. The name SAMBA is a variant of SMB, the protocol from which it stems.

2.2.3 State-of-the-Industry Review

File storage services have been widely implemented in the ICT industry. The services that run on a Linux-based environment provides world class performance, scalability, and availability for file storage systems. This is true since many international companies already put their trust on Linux-based file storage services. In addition, other ICT sectors, such as the education sector, have been using Linux file storage services in their daily operations. Several success stories from these sectors are worth discussing in the following subsections.

2.2.3.1 Indiana University

Indiana University (IU)[3] offers a scalable, network accessible, open, standards-based storage infrastructure to support teaching, research and administrative computing. This is accomplished by the means of a Linux-based Common File System (CFS) used for general data storage for IU campuses based on the Distributed Computing Environment Distributed File System (DCE DFS) and a global file system for remote collaboration and data sharing based on the Andrew File System (AFS).

The Common File System (CFS) service is a distributed storage service offered by Indiana University to the faculties, staff, graduate and undergraduate students who need a modest amount of data storage, but require seamless access to it from different research and instructional environments at IU. CFS is delivered using the Open Group's Distributed Computing Environment Distributed File System (DCE DFS)

[3] http://www.indiana.edu/

product.

The AFS service is offered to those researchers at IU who participate in inter-institutional collaborations that have chosen AFS as their primary means of data sharing. AFS is not intended for general data storage at IU. CFS is the appropriate services for this purpose. The Andrew File System can be accessed from some research hosts and from the UITS UNIX workstation clusters. Researchers can request installation of AFS clients on their desktop workstations and departmental servers.

2.2.3.2 Carnegie Mellon University (CMU)

Coda [10] is an advanced networked file system implemented at Carnegie Mellon University (CMU)[4]. It has been developed at CMU since 1987. Coda is a distributed file system with its origin in the CMU's AFS2 file system. It has many features that are very desirable for network file systems. Coda enables file sharing with strict security policies implemented in the system. It ensures that only authorized clients can have access to the file system. It has been a successful system with wide usage not only by CMU but it has been published to the public.

2.2.3.3 Chronicle Heralds

The Chronicle Herald[5] is the biggest independently-owned newspaper in Canada and boasts a daily readership of 333,000. As a thoroughly automated newspaper operation, The Chronicle Herald requires a highly available server and operating system to keep operations moving 24 hours a day. They recently migrated from a Sun Enterprise 3000 environment to an IBM eServer BladeCenter with Red Hat Enterprise Linux, a decision that saves CA$14,000 (approx RM 40,500) over three years in maintenance costs alone. Additionally, The Chronicle Herald uses SAMBA and Sendmail for file system and e-mail software rather than Microsoft Windows NT and Microsoft Exchange,

[4]http://www.cmu.edu
[5]http://www.canoe.ca/ChronicleHerald/home.html

creating a savings of CA$25,000 (approx RM75,000).

2.3 Printing Services

Printing is another important network service that must be made available in every computer networked environment. Printing services make networked printing peripherals and software easier to be used and managed. This subsection describes two options currently available to be utilized for printing management, namely Common UNIX Printing System (CUPS) and KDEPrint.

2.3.1 CUPS Technology

Common UNIX Printing System (CUPS)[6] is a set of applications that allow UNIX-compatible OSes to control printer devices, manage print queues and processes print requests. CUPS is based upon an emerging Internet standard called the Internet Printing Protocol (IPP) [12]. IPP has been embraced by dozens of printer and printer server manufacturers and is supported by Microsoft Windows 2000 and above.

IPP defines a standard protocol for printing as well as managing print jobs and printer options like media size, resolution, and so forth. Like all IP-based protocols, IPP can be used locally or over the Internet to printers hundreds or thousands of miles away. Unlike other protocols, however, IPP also supports access control, authentication, and encryption, making it a much more capable and secure printing solution than older ones. IPP is layered on top of the HyperText Transport Protocol (HTTP) which is the basis of web servers on the Internet. This allows users to view documentation, check status information on a printer or server, and manage their printers, classes, and jobs using their web browser. CUPS provide a complete IPP/1.1-based printing system that provides basic, digest, and local certificate authentication and user, domain,

[6] http://www.cups.org

or IP-based access control.

2.3.2 KDEPrint

KDEPrint[7] allows users and/or administrators, depending on their rights, to access printing subsystems (such as CUPS, LPD, RLPR, LPRng and PDQ) through a KDE[8] graphical user interface (GUI). Using KDEPrint, users and/or administrators can print, administer jobs, printers and the printing daemon, all in a comfortable manner. Different users may use different printing systems on the same box. A user is free to even switch on the fly, from the print dialog, the print subsystem to be used for the next job.

Most PC users are used to Linux Printing Daemon (LPD) printing. LPD provides only basic printing functions. It is very inflexible and does not utilize the many options of the more modern print systems like CUPS. While also working remotely over any distance (like every TCP/IP based protocol), LPD lacks bi-directional communication, authentification, access control and encryption support. KDEPrint can use CUPS to support, among others;

- querying the LAN for available printers;
- basic, digest, and certificate authentication;
- access control based on IP addresses, net addresses, netmasks, host and domain names; and
- 128-Bit TLS or SSL3 encryption of print data, to prevent eavesdropping (or at least make it much more difficult).

These make KDEPrint a much more robust and reliable solution than using the vulnerable LPD [7]. KDEPrint supports CUPS for system administration and on the client

[7] http://printing.kde.org/

[8] KDE is a powerful Unix Free Software graphical desktop environment for Linux and UNIX workstations. It combines ease of use, contemporary functionality, and outstanding graphical design with the technological superiority of the UNIX operating system.

side.

2.3.3 State-of-the-Industry Review

For years the printing problem has plagued UNIX [11]. Unlike Microsoft Windows or Mac OS, UNIX has no standard interface or system in place for supporting printers. Among the solutions currently available, the Berkeley and System V printing systems are the most prevalent. These printing systems support line printers (text only) or PostScript printers (text and graphics), and with some coaxing they can be made to support a full range of printers and file formats.

However, because each variant of the UNIX operating system uses a different printing system, developing industry-proven printer drivers for a wide range of printers and operating systems is extremely difficult. That is combined with the limited volume of customers for each UNIX variant, has forced most printer vendors to give up supporting UNIX entirely.

Fortunately, a solution in the form of CUPS has appeared in the industry. CUPS is designed to eliminate the printing problem. It is a common printing system can be used by all UNIX variants to support the printing needs of users. Printer vendors can use its modular filter interface to develop a single driver program that supports a wide range of file formats with little or no effort. Since CUPS provide both the System V and Berkeley printing commands, users (and applications) can reap the benefits of this new technology with no changes.

2.4 Linux Application Services

There are many applications available in the ICOSYS2 Linux-based platform. X is the main application used on the ICOSYS2 platform. It is a software that allows X-window applications to be drawn on a display. It runs a graphical interface for the user

similiar to the Microsoft Windows products, but X-servers can run multiple sessions on remote computers.

Remote application usage through the network can be done using several methods such as through terminal emulation software such as Telnet (for text based command line interface), X-server-based (such as using X-Win32[9] that pipes X-server interface onto Microsoft Windows desktops), and remote computing software (such as using Virtual Network Computing[10] (VNC) that loads the whole X-desktop onto Windows desktops).

Terminal emulation software can be used to execute text-based command line applications (CLI). It does not require a graphical interface to execute CLI programs. Such programs include text processors (pico, vi, and emacs), lynx (text-based web browser), and networking tools (such as traceroute, ping, and netstats). However, programs using the text-based approach is very inconvenient to use, because users have to write and remember every command or code execution script[5].

Alternatively, ICOSYS2 provides graphical-based applications that run using X-server software, for example, office application suites (StarOffice[11], OpenOffice[12], Lyx[13] and WordPerfect[14]), and web browsing (Mozilla[15], Galeon[16], Opera[17]). The quality of these applications is better or at least on par with that of their proprietary cousins.

As mentioned before, access to these applications from Windows (Mac, OS/2 or other platforms) can be made in two ways; 1) by using X-server-based software, such as using X-Win32 that pipes the X-server interface onto Microsoft Windows desk-

[9] http://www.starnet.com
[10] http://www.realvnc.com
[11] http://www.sun.com
[12] http://www.openoffice.org
[13] http://www.lyx.org
[14] http://www.wordperfect.com
[15] http://www.mozilla.org
[16] http://galeon.sourceforge.net
[17] http://www.opera.com

tops, and 2) by using remote computing software, such as Virtual Network Computing (VNC) that loads the whole X-desktop onto Windows desktops. We explain these alternatives in the following sections.

2.4.1 Virtual Network Computing (VNC)

VNC stands for Virtual Network Computing which is a remote control software which allows the user to view and interact with a target computer. VNC software makes it possible to view and fully-interact with one computer from any other computer or mobile device anywhere on the Internet. VNC software is cross-platform, allowing remote control between different types of computers. For ultimate simplicity, there is even a Java viewer, so that any desktop can be controlled remotely from within a browser without having to install new software. VNC has a wide range of applications including system administration, IT support, and helpdesks.

VNC can also be used to support the mobile user, both for *hot-desking* within the enterprise and also to provide remote access at home, or on the road. The system allows several connections to the same desktop, providing an invaluable tool for collaborative or shared working in the workplace or classroom. Computer support within the geographically spread family is an ever popular use. The downside of VNC is that it requires to load the whole operating system desktop onto the client desktop. This would require a large network bandwidth to carry the whole desktop (GUI) dump.

2.4.2 X-Win32

X-Win32 is the PC X-server application that enable connection to multiple UNIX/Linux systems at the same time, thus paving the way for running X-window applications remotely. It does not load the whole UNIX desktop, but only the applications selected by users. This minimizes the network load thus allowing more applications to be run at the same time. Our preference for X-Win32 is due to the fact that it exceeds other com-

peting PC X-servers in critical performance areas such as ease of use, performance, stability on all windows platforms, configurability, installation, functionality, price, online help, support and standard and advanced features (see Table 2.1).

2.4.3 State-of-the-Industry Review

PC X-servers are used to display graphical UNIX/Linux applications on Windows desktops. These programs assist administrators with the remote maintenance of heterogenic networks. Network Computing Magazine tested nine currently available PC X-servers: X-Win32 ver 5.0 by StarNet, View Now 1.01.3 by Netmanage, Kea 4.1 by Attachmate, Omni X 6.2 by Xlink, HOB Link X11 4.6 by HOB, Xvision Eclipse 7.3 by SCO, Exceed 6.2 by Hummingbird, Reflection Suite 8.0 by WRQ, and X-WinPro by Lab-Pro. Each X-server was prompted to open various single and multiple window sessions on the following four UNIX/Linux operating system platforms: AIX 4.3.2, SCO UNIXware 7.1.1, Solaris 8, Linux 2.2. From the test results, the winner of this Network Computing Magazine comparative test of PC X-servers is StarNet's X-Win32 5.0 (Table 2.1 shows the Interactive Report-Card). StarNet's X-Win32 5.0 wins because it combines function, speed and operational ease quite well compared the other PC X server software. X-Win32 5.0 is compact, fast, and simple to operate. It offers a very good representation of all UNIX systems, except for some fonts under CDE or KDE. Linux desktops were also displayed perfectly in version 5.0 as were all UNIX desktops.

In addition, X-Win32 is a commercial application made available for free to all North Carolina State University (NCSU) faculties, staff, and students through licensing arranged by the College of Engineering and Starnet Communications. The program allows Windows users to connect to Linux/Solaris (or other UNIX-based) systems on a network and display the windows from applications running on those systems back to their Windows machines.

Features	**Weight**	A	B	C	**D**	E	**F**	**G**
Installation	20%	4.5	4.5	4	4	4.5	3.5	3
Functionallity	20%	4.5	4.5	4.5	3.5	3.5	4	3.5
Ease of Use	20%	5	4.5	4	4	3.5	3	4
Configurability	10%	4.5	4	3.5	4	3.5	3.5	3.5
Tools	10%	3.5	4.5	4	4	4	4.5	4.5
Price	10%	4.5	4	4	4.5	5	3	3.5
Performance	10%	4.5	4.5	4	4.5	3.5	5	3
Overall	100%	4.5	4.4	4.0	4	3.9	3.7	3.5
A>4.3, B>3.5, C>2.5, D>1.5 Rating A through C are in order		A-	A-	B+	B+	B	B-	B-

Legend:
(A) Starnet X-Win32 5.0
(B) SCO Xvision-Eclipse 7.3
(C) Attachmate Kea 4.1
(D) WRQ Relfection Suite for X8
(E) Lab Pro X-WinPro
(F) Hummingbird Exceed 6.2
(G) Netmanage View Now 1.01.3

Table 2.1: Interactive Report-Card: PC-X-Server Source [8]

X-Win32 was rated as the industry's best PC X-server in the most comprehensive objective review of leading PC X-server in more than five years. From the outset in 1989, X-Win32 has remained focused on the X-server functionality, high performance, ease of use, stability and low cost. That makes X-Win32 the ideal remote host access server for today's customers [1]. For this reason, we chose X-Win32 for supporting our ICOSYS2 application service.

2.5 Backup and Retrieval Services

The baseline motivation behind all backup systems is disaster recovery. All backup technologies meet this goal by making a copy, but there are really two kinds of copying, with distinct characteristics: archival, and mirroring [6]. Archival backup gives the ability to travel through time; the user suddenly realizes that an important file is missing, and he/she is not sure when it was deleted, then the ability to shift through a

year of backup dumps looking for the missing file can be a life-saver. In order to do this, however, the user must keep a lot of data stored around, and that almost always means putting the backup dumps on some sort of offline storage, such as CD-R (or CD-RW), Zip, etc.

Mirroring backup gives the user immediate access to the most recent copy of their data; if the user deleted that important file just this morning, then it is a simple matter of retrieving it from the backup drive, without any searching. (If the user deleted it two days ago, chances are that he/she is already out of luck.) At a minimum, mirroring only requires a spare disk of comparable size, and is easy to automate completely, as it requires no manipulation of offline media.

The three *entry-level* backup options for Linux (and UNIX systems generally) tend to provide either archival or mirroring, but not both. They are;

- The standard UNIX *tar* utility. This is GNU *tar* in free implementations (and even some other UNIX flavors), and is the easiest tool for archiving particular directories. It has the distinct advantage of being supremely portable; *"tar"* format can be read by all other UNIX systems, and even by Windows and Macintosh.

- The traditional UNIX *dump* and *restore* programs. Most Linux systems come with the *dump/restore* implementation for *ext2/ext3*, but these are the traditional names for the archival backup programs in UNIX, so "*man dump*" will almost always come up with something on any UNIX system. *dump* can only operate on a whole partition at a time, but it features incremental backup capability, so we only need to back up what's changed.

- The *rsync* program. Unlike *tar* or *dump*, *rsync* is designed to mirror the content of directory trees over the network, is quite clever about only transferring data that have changed, and can also be set up to do local disk-to-disk copying.

In order to reduce the amount of storage required for archival backups, it is desirable

to skip files that have not changed since the previous backup. Obviously, the first backup must contain everything, but a series of subsequent backups need only contain the changed files since the last backup; in the event of a disaster, restoring all backups in the order in which they were made will return the file system to the same state as if it had been restored from a single full backup made on the last day. This scheme still has two drawbacks; the first is that the process of restoring the file system gets to be quite tedious after a few weeks worth of backups, since there are quite a few of them at that point. Worse yet, the second drawback is that data will be lost if any of those backups somehow got lost or corrupted.

2.5.1 Backup levels

In order to address these drawbacks, it is useful to define a backup level between 0 and 9 that controls how comprehensive to make the backup. Each level k backup contains a snapshot of all files changed since the level k-1 dump (or the dump made at the next lower numeric level if there is no level k-1 dump). Level 0 is therefore the most comprehensive, and level 9 is the most incremental. At this point, some additional terminology is in order:

A full backup dump is a complete snapshot of the state of the filesystem at the time the backup was made. Full dumps are defined to be at level 0. A consolidated backup dump is a snapshot made at a later time that only contains those files that have changed since the last full dump. Consolidated dumps are always at level 1, and are usually made weekly. When these get to be too large, then it's time for a new full dump. Incremental dumps have levels between 2 and 9 inclusive, and are usually made daily. A given incremental dump may or may not reach all the way back to the last consolidated backup, but it certainly will not reach any farther. In order to reduce the number of incrementals required, one can use the modified "Tower of Hanoi" algorithm [2] described in the dump manpage, which prescribes the following sequence of dump

levels:

3 2 5 4 7 6 9 8 9 9 ...

These are for daily backups, which is the absolute minimum period for a workgroup server in an office environment. At the end of the week, a consolidated dump is performed, and the daily cycle starts over again. At this point, last week's incrementals could be thrown away, as they are no longer needed for disaster recovery, but it's a good idea to keep them around for at least a month in order to cover the "I didn't mean to delete that" syndrome. In any case, this multilevel backup system turns out to be quite effective in reducing the size of backups; even after a month, a consolidated dump can be only about 20% of the size of the full dump, and the incrementals only 3 to 5%.

2.5.2 State-of-the-Industry Review

Backup is a required service bacause of the nature of secondary devices that are somewhat prone to failure. Backup is a tertiary option for storage of the *old* data which may be needed later in the future. As a backup solution for the open system, Linux provide a wide option of backup mechanism. In the industry Linux has attract a lot of major and not so major companies have to switch from proprietary system. Companies like Precision Steel, Veritas and Oracle has already invested in open system based solutions. We describe some of the related industries case studies below.

2.5.2.1 Oracle

Oracle[18] is the largest database company in the world and a force behind the e-commerce revolution. They protect 35 terabytes of data, spanning hundreds of servers in a heterogeneous, multi-data center environment with open system backup solution. Globally,

[18]http://www.oracle.com

the company has over 690 servers, 250 of which are enterprise-class machines, primarily open platforms. The main data center, at the companys Redwood Shores headquarters, supports 24,000 U.S. employees, and has 18 terabyte of production data. The Redwood Shores data center has 158 production servers, mostly UNIX. This server environment consists mainly of Sun enterprise class midrange machines including the Sun Enterprise 3000, 5000 and 6000. There are also Hewlett Packard 9000 K-class and T-class servers, primarily for Oracle's the Support Services group.

2.5.2.2 West Chester University

West Chester University[19] in West Chester, Pennsylvania currently serves nearly 12,000 undergraduate and post-graduate students. However, the IT department at West Chester University is dealing with the very modern issues surrounding backup and restore processes for their rapidly changing IT infrastructure. SANs can be used to improve backup and recovery performance, as well as minimizing disruption to applications today. West Chester University of Pennsylvania was able to relieve their backup burden by carefully implementing a LAN Free backup SAN in stages. With the implementation of the open system SAN solution, it manages to serve all the users efficiently.

2.5.2.3 Precision Steel

With facilities in Illinois, California, North and South Carolina, Texas, Washington, Canada and the United Kingdom, Precision Steel[20] processes countless customer orders in a largely distributed environment. Quality-control policies and excellence-in-manufacturing processes form the backbone of company operations, with each incoming vendor shipment carefully recorded and tagged for size, grade and consistency. A series of decisive inspections assures product quality in a variety of alloys. As a model of innovation in the steel service-center industry, Franklin Park, based in Pre-

[19] http://www.wcupa.edu
[20] http://www.precisionsteel.com/

cision Steel Warehouse, Inc. (Precision Steel), combines expertise with efficiency to optimize both the customer experience and company production. Adopting an open system backup solution is the most cost effective solution for the company.

2.6 ICOSYS2 Implementation Strategy

Our method to implement ICOSYS2 services is based on the Microsoft Windows 2000 Test Lab strategy. Althrough the strategy is meant for implementing a proprietary system, it can suitably be adjusted for implementing an open source computing environment. Our strategy determines how the services are implemented based on it's surrounding environment and users' needs. It is important to understand the current environment and determine the best suitable components. There are three steps in this strategy namely 1) determine supported hardware and software, 2) design logical network and 3) service implementation.

2.6.1 Determine Supported Hardware and Software

2.6.1.1 The Operating System

The implementation of ICOSYS2 is based on an open source system. For ICOSYS2, Linux (using Red Hat version 9) has been chosen as the server operating system. Redhat 9 is one of the popular Linux distributions. It is distributed by Redhat [21] Corporation and by far it is the most comprehensive version of Linux.

The Linux kernel is originally written by Linus B. Torvalds under the terms of the General Public License (GPL). The GPL states that the source code must be freely distributed and anyone is allowed to make copies for their own use, or to sell or give to other people (with a few restrictions). While most Linux software is under GPL, this does not mean that all software developed or ported to Linux has to be. Many

[21] http://www.redhat.com

Item	Type
Processor	Intel Xeon 2.4 GHz
Memory	1 Ghz DDRAM
Hard Disk	IBM SCSI 120 GB
Graphic Card	ATI RAGE XL

Table 2.2: Server Hardware Spesification

other licenses exist, with some commercial software packages having more restrictive licenses, such as the common copying restrictions faced by Windows users. However, the GPL allows other users to utilize the system, change it and distribute it.

2.6.1.2 The Hardware

The hardware used to support the ICOSYS2 services has been considered from several aspects such as task workload, processing capacity, printing capability, users support, funding and logical network design. We have purchased an Intel Xeon 2.4 Ghz processor-based server with one GByte of memory. The server is equipped with 120 Gbyte of SCSI storage capacity (table 2.2).

For printing services, one HP LaserJet 2300n has been acquired which comes with network printing capabilities. The hardware configuration should be enough to support more than 240 FTM users, with each user is provided with 500 MByte of storage space. One gigabyte of memory may cater for 50 users, with 10 MByte of memory each. We believe that the resources is enough for each user, for this specific purpose.

2.6.2 Logical Network Design

Logical network design includes the conceptual configuration or design of the network devices on a network and how they are interconnected. Based on current FTM network topology, we have added the following devices to the network; a server and a printer (see Figure 2.2) . The FTM logical network is divided into three levels, with each

level connected to one or two Ethernet switches (Avaya's P333 switches). Each level switches are then connected to the main switch (Avaya's P550) to form the local area network for FTM. The servers and printers are located on Level One where the Faculty administrative office is located.

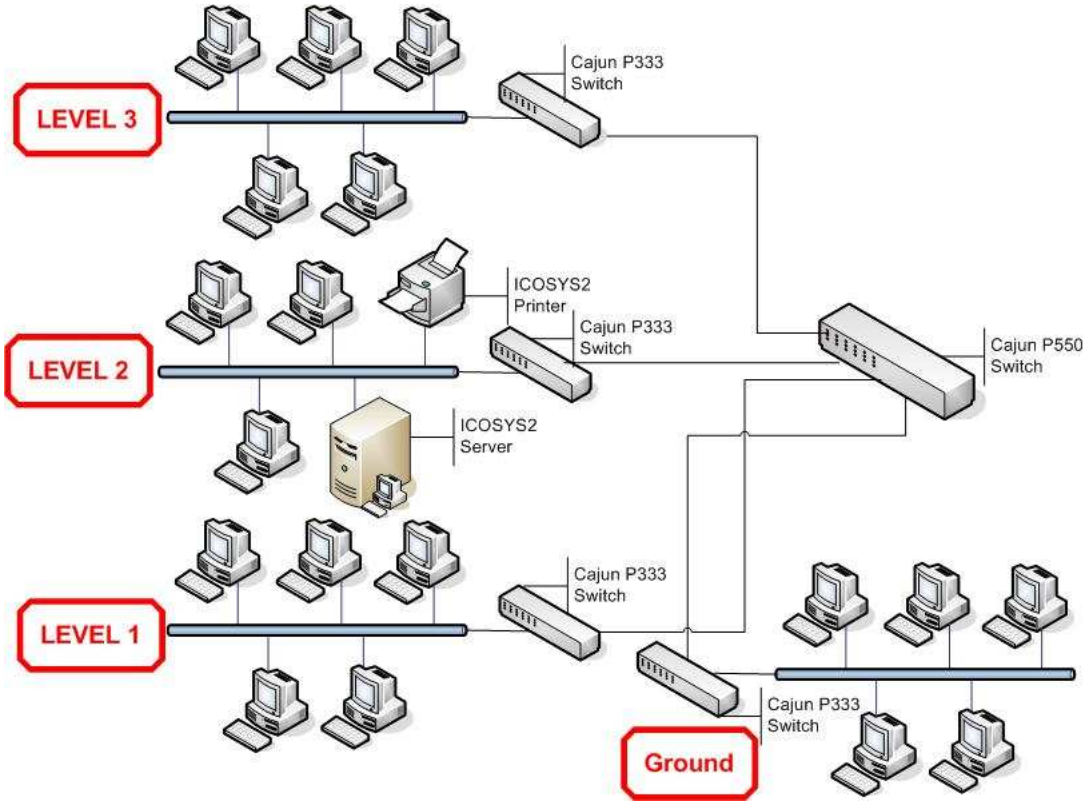


Figure 2.2: ICOSYS2 Logical Design

2.6.3 Services Implementation

Since ICOSYS2 will serve as a basic computing environment for FTM, the design of the services must also take into consideration the current scenario and the implementation plan. We have divided the ICOSYS2 implementation tasks into four different parts which represent each of the services, as outlined below. Further discussions on each services is provided in the subsequent chapters.

2.6.3.1 File

File storage services basically provide users with the ability of centrally storing and sharing data. A distributed file system stores files on one or more computers called servers, and makes them accessible to other computers, called clients, where they appear as normal files.

2.6.3.2 Application

Any ICOSYS2 open source application can easily be accessed from local computers that can run simultaneously. Users can use a proprietary operating system to access the applications remotely without having the need to install them on their machine.

2.6.3.3 Printing

ICOSYS2 printing services allow users to send their print jobs to the respective printers across the FTM network. Printing can never be easier with the implementation of this service. Users can access their file virtually from anywhere and control their printing remotely without much hassle.

2.6.3.4 Backup

The ICOSYS2 project delivers the benefit of having a centralized file and application storage and sharing environment for FTM citizens. Nevertheless, it is indeed not a wise decision for users to be fully dependent on the ICOSYS2 server to provide a secondary backup solution. The server itself is prone to failure. For this reason, we needs a tertiary means to backup the user's data.

2.7 Summary

This chapter has provided the background of our ICOSYS2 research. An overall description of each of the tasks by outlining them in the state-of-the-industy as well as the technological perspectives was provided. The ICOSYS2 implementation stategy that serves as our roadmap for this project was also outlined.

The subsequent chapters provide further discussions on the implementation of the ICOSYS2 services.

Chapter 3

ICOSYS2 Files Storage Services

3.1 Introduction

This chapter describes our detailed implementation of the ICOSYS2 network file storage services. Section 3.2 outlines the service requirements of the services. Section 3.3 details the implementation of the ICOSYS2 file storage services. Section 3.4 summarizes the chapter.

3.2 Service Requirement

Currently, the FTM workstations are of stand-alone nature, relying on a single, Microsoft-based platform environment. Most of these machines run Microsoft Windows-based systems (Win9x/2000/XP). User's work (mainly academic and administrative) are kept in these workstations. There is no centralized file management system and no backup services provided. It mainly depends on the individual user whether or not he/she wants to have his/her own data to be backed up on some other secondary devices. This is not a good practice since trusting a single device is precarious. If anything happens to the primary storage such as disk failure or virus attack, all the hard work is gone and there is no way to retrieve them back.

In this environment, file sharing is mostly based on peer-to-peer arrangement. Client-server option for sharing files has never been explored in FTM. Under this peer-to-peer arrangement, the user must first turn on their machine, then enable the sharing properties of the desired folder. The files cannot always been made available for sharing because of the uncertain availability of each individual machine. For example, a workstation might be turned on or off by its owner. Thus, the accessibility to the files can only be granted while the machine (within which the files reside) is on.

The ICOSYS project attempts to introduce, culturalize thus provide and alternative solution for FTM citizens. Under ICOSYS, the client-server option for file sharing is introduced. These file storage services offer FTM users with the ability of centrally storing and sharing data. The usefulness of a distributed file system becomes clear when considering a group of FTM staff sharing documents.

There are several advantages to this arrangement. The files are more widely available since many computers can access the servers, and sharing the files from a single location is easier than distributing copies of files to individual clients. Users can save and retrieve files from anywhere (initially within FTM) with normal functionality, like assessing his/her local disk. Besides file sharing, applications can also be shared under this arrangement (details of which is covered in Chapter 4). Additionally, backup and safety of the information are easier to manage since only the servers need to be backed up. In both cases, system administration becomes easier.

3.3 Service Implementation

The ICOSYS2 file and storage services are implemented using SAMBA[1]. SAMBA is a collection of programs that use the SMB protocol to allow sharing of Linux/UNIX file systems and printers as well as to allow sharing of Windows client file space and printers with Linux/UNIX systems. In practical terms, SAMBA allows user to turn a

[1] http://www.SAMBA.org

Linux machine into a file server on a Windows network [13].

The two key program files of the SAMBA suite are SMBD and NMBD. SMBD is the server daemon that provides file sharing and printing services to Windows clients. The server provides file space and printer services to clients using the SMB protocol. NMBD, on the other hand, is a server daemon that understands and can reply to NetBIOS over IP name service requests. NMBD participates in the browsing protocols which make up the Windows "Network Neighborhood" view. SMBD performs most of the SMB services and NMBD provides NetBIOS Windows with name service that allows other computers to browse resources over the FTM network. The implementation of the ICOSYS2 file storage services is discussed in detail in the following subsections.

3.3.1 SAMBA Installation

SAMBA is normally available as precompiled binaries in the Redhat 9.0 and other Linux distributions. In this project however, we have decided to use the latest version of the software. For this reason, the best place to look is in the SAMBA's official web site [2].

SAMBA is an open source application under the GPL. SAMBA is normally distributed in two packages; binary and source files. The binary package is a setup program for SAMBA where users can easily install the file using a package installer, whereas the source files require users to compile the source code into a binary executable file.

The version of SAMBA that we used in this project is version 2.2.8 (as of 20 September 2003). We chose to install SAMBA using the binary package. The package is easy to be identified with their **.rpm* file extension. The installation step is conducted using Redhat 9.0 Linux distribution[3]. The steps taken to install SAMBA

[2] http://www.SAMBA.org

[3] Other Linux distributions might require some extra steps.

are;

- Download latest version of SAMBA from the official SAMBA website.
- Pop up a terminal window and change directory to the downloaded SAMBA installation file.
- As root, run the installation command from terminal window:

```
[root@ICOSYS2]# rpm -Uvh SAMBA-2.2.8a-2_rh9.i386.rpm
```

3.3.2 Configuring SAMBA

Most of the SAMBA settings are contained in a configuration file called `smb.conf`. By default, this file is placed under "`/etc/SAMBA`" directory. The `smb.conf` file determines how the SAMBA server works and behaves. To manually configure the SAMBA configuration file is tricky - one must know what one is doing. Misconfiguration of the file can cause havoc to the system.

The configuration file contains three special sections: `global` (for server setting), `homes` (shared directories), and `printers` (printing service) which can be identified with square brackets (refer Figure 3.1). We can also configure `smb.conf` using third party software like Webmin[4] or SWAT[5]. Webmin is a web-based application that provides good manipulation over `smb.conf` without screwing up the files. In this project, we manually configured the file.

[4] freely available from http://www.webmin.org.
[5] SWAT is part of the SAMBA suite.

```
[global]

   workgroup = MYGROUP
   server string = Samba Server
;  hosts allow = 192.168.1. 192.168.2. 127.
   printcap name = /etc/printcap
   load printers = yes
   printing = cups
;  guest account = pcguest
   log file = /var/log/samba/%m.log
   max log size = 0
   security = user
;  password server = <NT-Server-Name>
;  password level = 8
;  username level = 8
   encrypt passwords = yes
   smb passwd file = /etc/samba/smbpasswd
;  ssl CA certFile = /usr/share/ssl/certs/ca-bundle.crt
   unix password sync = Yes
   passwd program = /usr/bin/passwd %u
   passwd chat = *New*password* %n\n *Retype*new*password*
   pam password change = yes
;  username map = /etc/samba/smbusers
;  include = /etc/samba/smb.conf.%m
```

Figure 3.1: Some of the content of smb.conf

In Figure 3.1, we make changes to two important parameters in the `global` section, namely, `netbios name` and `workgroup`. We set the `netbios` name to the name of the server and the workgroup name to the name of the workgroup within which the server is located. In our case the netbios name is `ICOSYS2` and the workgroup is `ftm`.

3.3.3 User Setup

For each ICOSYS2 user, the default their home directory is `/home/[username]`. For example, when a user registers his name as `ahmad`, the ICOSYS2 server will automatically create a directory called `/home/ahmad` as the user's home directory. Users can then access their home directory if the *homes* section (in `smb.conf`) is configured.

At least four parameters are involved in sharing the user's home directories. The first parameter is `path`, which is to determine the user's shared directory. The `valid`

`user` parameter determines the list of users that will be allowed to access the files. The `Browseable` parameter allows user to browse folder and files in the shared directory, and the `writable` parameter determines weather or not a user can write to the disk.

For each user, the following script must be inserted in the `/etc/SAMBA/smb.conf` file.

```
[username]
path = /home/[username]
valid users = [username]
browsable= Yes
Writable= Yes
```

After creating the user identification profile for the SAMBA server, each user has to set his/her password for accessing the SAMBA server. To set the user password, a user has to log in to the server and then type `smbpasswd <username>` at the console and enter his/her SAMBA password. The password can be different from the user login password.

3.3.4 Testing and Starting the SAMBA Server

The SAMBA server needs to be running in order for the change to take place. To start a SAMBA service type `service smb start` in a terminal window. If there is no problem with the configuration file and the network connection, the following message will appear.

```
[root@ICOSYS2]# service smb start
Starting SMB services:  [ OK ]
Starting NMB services:  [ OK ]
[root@ICOSYS2]#
```

The SAMBA server can now successfully start using the modified configuration file.

3.3.5 Mapping Network Drives from Windows Clients

Mapping network drives from a Windows-based client to the SAMBA server depends on the Windows version the user is using. For Windows 95/98/Me, the setup for mapping will require the user to ensure that the computer name and username for the computer is similar to the username registered with the SAMBA server.

However, Windows 2000/XP users do not have to change their computer name. Just by mapping their network drive, users will have the same look and feel as accessing their local file system, as depicted in figure 3.5.

3.3.5.1 Mapping Network Drive From Windows9x (95/98/Me) Clients

Mapping ICOSYS2 network drives from Windows9x clients require users to change their Windows computer name into the username registered in the SAMBA server. To change computer name from Windows9x;

- go to Windows *control panel*,
- select *network* option (see Figure 3.2),
- select the *Identification* tab,
- change the computer name to the one that is registered in the SAMBA server, and
- restart the computer for the change to take place.

After restarting the computer, mapping the Windows network drive to the ICOSYS2 drive can be done by opening Windows Explorer. From Windows Explorer, select *Tools* menu from the drop down menu and then select the *Map Network Drive* option. Select the drive the user wants to use and then put the server path location using

`\\servername\username` (see Figure 3.3). Then the user will be asked for the password (i.e., the SAMBA user login password).

Figure 3.2: Changing WindowsComputer Name

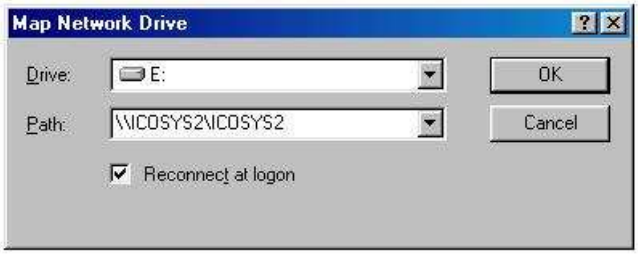


Figure 3.3: Map Network Drive

3.3.5.2 Mapping Network Drives From Windows (2000/XP) Clients

The process of mapping network drives from Windows 2000/XP clients is much simpler. Users can easily map network drives using the *Map Network Drive* menu under

the *Windows Explorer* or *My Computer* (i.e., select *Tools* menu from the drop down menu and then select the *Map Network Drive* option). By mapping the network drive from a Windows XP/2000-based client, a user will just have to set the drive letter with the server path and then enter a valid SAMBA username and password (see Figure 3.4).

Figure 3.4: Map network drive from Windows XP

Figure 3.5: The Look and Feel of a local file system

3.3.6 Setting Disk Qouta

Each user is allocated with some amount of disk space for storing his/her files. To set the disk quota for each user, the Linux partition must first be allowed to support disk quota allocation. This can be done by editing the `fstab` configuration file which is located in the `/etc/fstab.conf` file.

Figure 3.6 shows the edited `fstab` file where "/" partition (boxed) changes from `default` option to `usrquota`. To set the disk quota for each user, `edquota <username>` command can be used. Then, edit the `fstab` configuration to add users.

Figure 3.7 shows the example of a user quota edited using the `edquota` command. The editing of user quota involves soft and hard block limits (which, initially each user is allocated with 200 MByte disk space).

```
[root@icosys2 root]# more /etc/fstab

LABEL=/                 /                       ext3    usrqouta        0 1
LABEL=/boot             /boot                   ext3    defaults        1 2
none                    /dev/pts                devpts  gid=5,mode=620  0 0
none                    /proc                   proc    defaults        0 0
none                    /dev/shm                tmpfs   defaults        0 0
/dev/hda3               swap                    swap    defaults        0 0
/dev/cdrom              /mnt/cdrom              iso9660 noauto,owner,kudzu,ro 0
0
/dev/fd0                /mnt/floppy             auto    noauto,owner,kudzu 0 0
[root@icosys2 root]#
```

Figure 3.6: `fstab` File

```
Disk quotas for user icosys2 (uid 571):
Filesystem           blocks       soft       hard     inodes     soft     hard
/dev/hda2             54244     204800     204800       2670        0        0
```

Figure 3.7: `edquota` Command

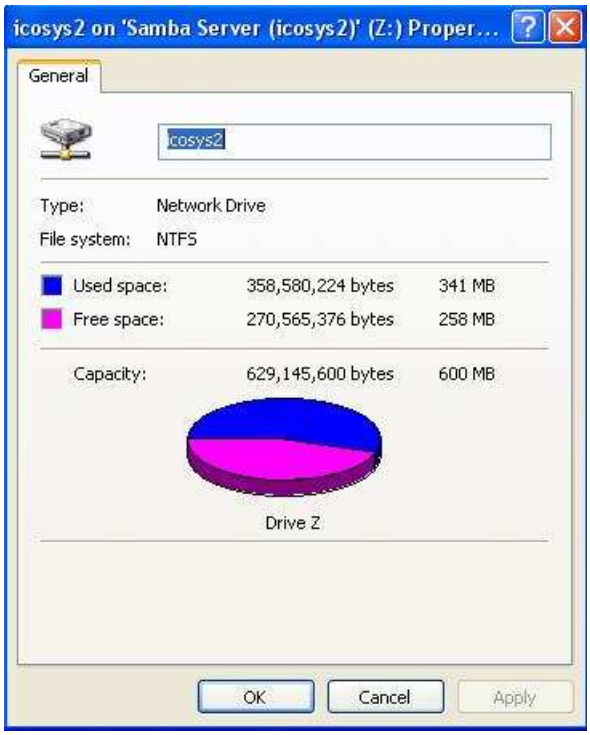


Figure 3.8: SAMBA Disk Space

3.4 Summary

ICOSYS2 file storage services provide FTM users with the capability of seamlessly sharing the file system. ICOSYS2 offers the ease of integration with other non-UNIX operating systems, where the non-UNIX users will not be aware that they are actually accessing an open source file system service. The look and feel is like the users are working on their own Windows-based system.

The integration of such service in ICOSYS2 can provide better file protection for FTM users where they can save and backup their data on the network storage. Users can directly access their directories with the convenience of browsing their local system. The implementation of ICOSYS2 network storage service is implemented using SAMBA. Configurations for server and client portions needed for this implementation have been explained in detail in this chapter.

Chapter 4

ICOSYS2 Application Services

4.1 Introduction

Applications help users accomplish their computing jobs. The open source applications are normally made freely available to users. Nowadays, open source applications are increasingly available and getting better or at least on par with those on-the-shelf applications. This chapter describes the implementation of the ICOSYS2 open source application services. Section 4.2 outlines the service requirements of the ICOSYS2 application services. Section 4.3 describes the implementation of the ICOSYS2 application services. Section 4.4 summarizes the chapter.

4.2 Service Requirement

The utilization of the open source software (OSS) is increasing globally. The OSS is often viewed as a significant change agent in the software industry. It has drawn much interest from proprietary software developers, lobby groups, industry analysts and regulatory bodies. In the process, governmental organizations, from the local to the national level, are considering or proposing rules or providing guidance in an effort to "level the playing field" between OSS offerings and those of proprietary providers.

The Malaysian Government, for example, has set up a task force to formulate and eventually launch its own National Open Source policy. It is for these reasons that FTM must assume proactive steps in adopting the open source platform in order to become a key player in utilizing (and eventually developing) the OSS. Hence, FTM must introduce open source system platform such as the Linux-based environment that can be used to help and support academic and administrative activities.

Nevertheless, most of the FTM staff are putting too much reliance on the Microsoft-based platform environment. The use and access to the OSS is very limited in FTM. Microsoft-based applications, such as the Microsoft Office suite, are normally installed on each user workstation. The process of installing and maintaining these applications on each staff workstation is both time consuming and inefficient. Moreover, the University has to spend a large amount of money to buy the campus license for these applications.

The ICOSYS2 project attempts to introduce, provide and thus inculcate the culture of using the OSS among FTM citizens. Under ICOSYS2, the client-server option for open-source application sharing is introduced. The ICOSYS2 server stores common OSS applications, and makes them accessible to other clients' computers via the FTM's high speed local area network. The application services offer FTM users with the ability of sharing not only applications but also their data (as described in the preceding Chapter 3).

The usefulness of this arrangement is clear when considering the reduction of effort needed in purchasing, installing and maintaining the needed applications. With ICOSYS2 application services, the open-source software can be obtained free-of-charge and is installed in a centralized server where access and administration tasks are made easier to perform.

4.3 Service Implementation

The implementation of the ICOSYS2 application services is focused on facilitating Windows-based users and enabling them to run OSS applications on their local machines. There are several mechanisms that can be used in running the OSS applications on Windows such as accessing through terminal emulation (e.g. by using Telnet or SSH sessions) and the X-service emulation (e.g. by using the VNC viewer or X-server software).

Accessing the ICOSYS2 applications via terminal emulation is not difficult since terminal emulation software is commonly available in the Windows environment. We provide further discussion on this issue in Subsection 4.3.2.

In providing ICOSYS2 services, we focused on the X-server-based applications. For this, ICOSYS2 offers two ways to access the applications. One way is by accessing the server applications using an X-server software (such as StarNet's X-Win32 or Hummingbird's Exceed) installed on the user's workstation. The other way is by using a remote control software (for example, VNC[1]) which allows users to view and interact with the ICOSYS2 server. The remote control software essentially dumps the whole Linux desktop interface onto the client desktop over the network. We believe that if too many of these applications are running simultaneously, they will consume much too bandwidth, thus degrade the network performance.

For this reason, we recommend users to choose the first option, i.e. using X-server software installed on the user's workstation. For this purpose, we encourage users to use the StarNet's X-Win32. Using X-Win32, only a portion of the Linux desktop (i.e. the Linux GUI) is piped through to the user's desktop. Hence, it requires a much lighter network bandwidth consumption, thus more convenient for user.

The implementation of the ICOSYS2 application services is discussed in detail in the following subsections.

[1] http://realvnc.com/what.html

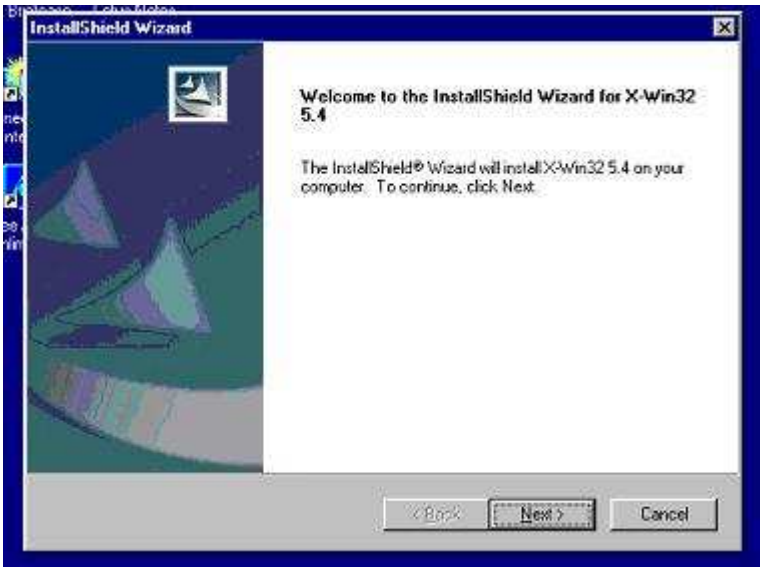


Figure 4.1: X-Win32 Installation Wizard

4.3.1 Installing and Running X-Win32

X-Win32 is the X-server application that allows users to run Linux's X-Windows graphical user interface on Windows machines. This software facilitates the Linux application that is selected by the user, to be run on the user's machine. The procedure to install and run X-Win32 is rather straight forward. The version we suggested for X-Win32 is version 5.4. The installation begins by choosing the desired language for the installation, then follow the on-screen wizard that guides the user through the entire installation process (see Figure 4.1).

After the installation is completed, X-Win32 can be run from the Windows *start* menu. Once invoked, the *"Connect Method"* window (see Figure 4.2) will appear. Simply click the option *Cancel*. Then, the main menu of X-Win32 will appear (see Figure 4.3). Once X-Win32 has started, an icon will appear in the Windows system tray, usually located in the lower right-hand part of the desktop. From this point, minimal configuration is required for X-Win32 and the main idea here is to run this Windows application in the background.

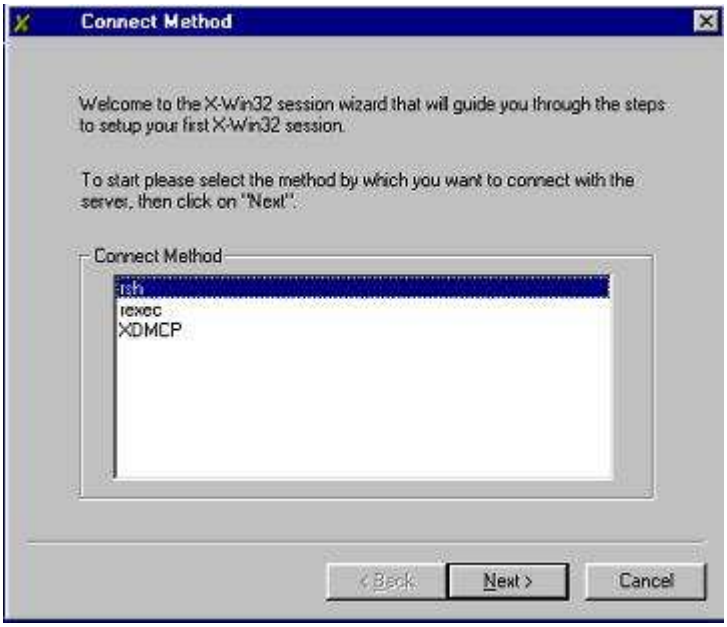


Figure 4.2: Connect Method

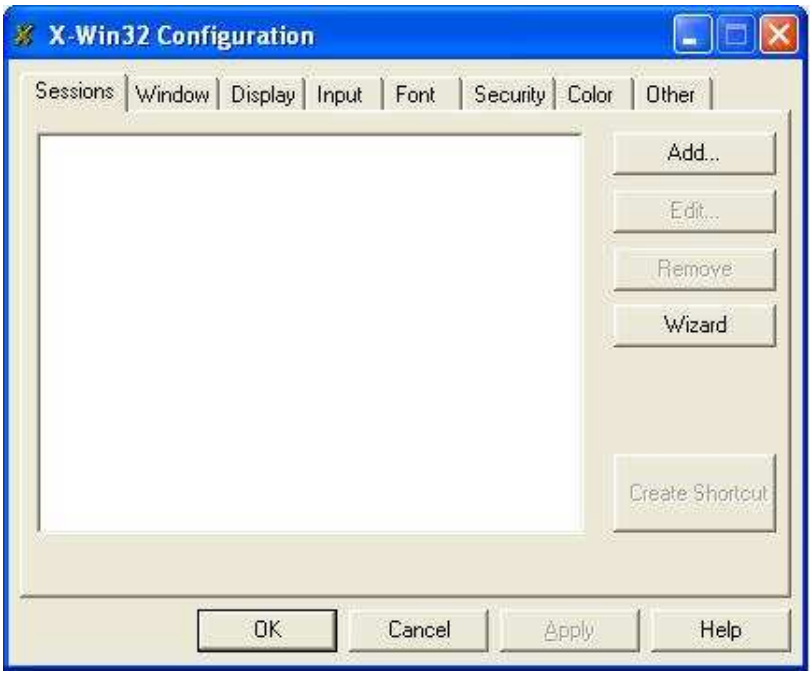


Figure 4.3: X-Win32 Main Interface

Figure 4.4: Command Prompt Under WinXP

Figure 4.5: Telnet Application

4.3.2 Running Remote Access using Telnet

To begin running Linux applications, a Telnet (or *ssh*) session has to be started first. The Windows Telnet application can be accessed through the Windows command prompt (see Figure 4.4). To get the Windows command prompt, click Windows *start* menu and click run `cmd` command (in Windows2000/XP) or `command` (in Windows9x). From the Windows command prompt, the Telnet session to the ICOSYS2 server can be started by typing `telnet icosys2`.

The client will then be connected to the server through Telnet application. The pop-up window of the Telnet session will appear as seen in figure 4.5. The user can then log in using the given username and password. A sucessful login to the system

results in a Linux command prompt as shown below - which shows that a telnet session has been successfully established:

```
[<username>@icosys2 <username>]$
```

After a successful login, the user must first execute the `export` command. To run this command the user need to know his/her machine IP address. Pop-up a Windows command prompt and run the `ipconfig` command (in Windows2000/XP) or the `winipcfg` (in Windows9x). The IP number information will appear in the dotted quad notation. After acquiring the user IP address, `export` command can be executed. From the terminal window, execute the following command:

```
[user@icosys2]$ export DISPLAY=<UserPC-IPaddress>:0.0
```

For example:

```
[user@icosys2]$ export DISPLAY=172.26.7.112:0.0
```

The purpose of the `export` command is to repipe the graphical display from the server to the designated user workstation. It is run in conjunction with the X-Win32 application that enables Linux graphical applications to be run locally on the user's machine. The Linux application that runs on the user workstation via this method will have the look and feel as if they were running on a Linux machine.

4.3.3 Running Graphical Linux Applications: An Example

There are many Linux OSS application packages available on the ICOSYS2 server. To run these applications, just type the name of the application from the Telnet window. Example of applications are listed in Table 4.1 below;

Applications	Description	Command
LyX	Document Processor	`lyx&`
OpenWriter	Word Processing	`ooffice&`
KDE	K Desktop Environment	`kicker&`
GBib	Bibliographic Application	`gbib&`
Open Draw	Graphic Manipulation Software	`oodraw&`
Open Impres	Presentation Software	`ooimpress&`
Galeon	Web Browser	`galeon&`
Mozilla	Web Browser	`mozilla&`
X-PDF	PDF viewer	`xpdf&`
GV	Post Script viewer	`gv` or `ggv`
GIMP	Image Manipulation Software	`gimp&`
NAM	Network Simulator Front-end	`nam&`

Table 4.1: List of Several Linux Based Applications

The command:

```
[user@icosys2 user]$ lyx&
```

will run LyX document processor, as in Figure 4.6.

The use of "`&`" sign, placed after the name of application, is to enable the application to run in the background. Hence, the same terminal window is able to run multiple Linux applications. In this way, a user can open multiple applications using the same terminal window.

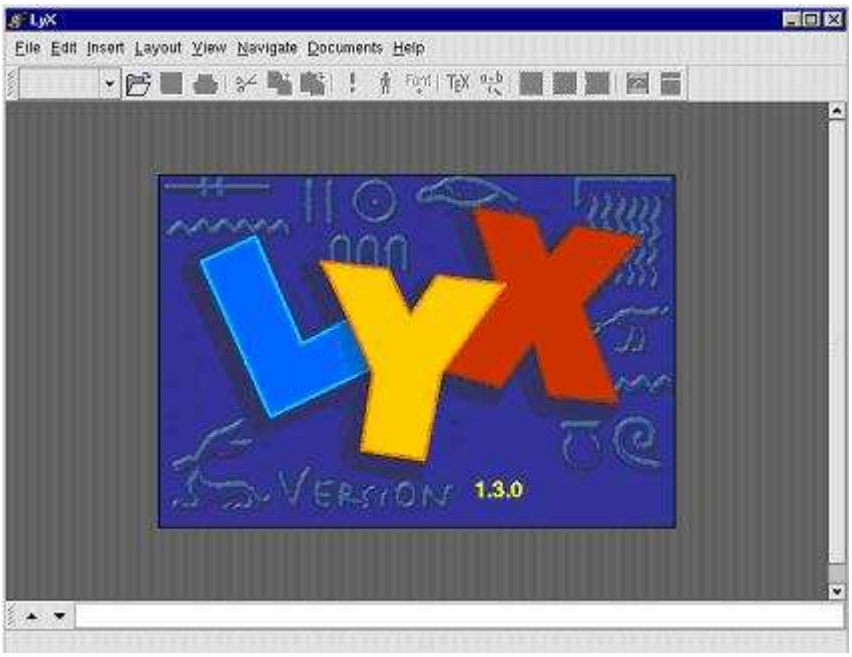


Figure 4.6: Lyx Application

4.4 Summary

The ICOSYS2 open source application services provide FTM users with the capability to execute open source software without the need to install them locally on the user's machines. All users can use the same software without having to worry about issues like license restriction and installation management.

The implementation of ICOSYS2 application services offers the opportunity for FTM users to utilize Linux applications. Users can now execute Linux applications from their Windows-based machine as if they are running local applications. Users can now have more applications that could not berun on their machines previously.

The implementation of ICOSYS2 application services is almost transparent to users. Users will not need to change their operating system while working if they wanted to run a Linux-based application. It is like having two operating systems in one machine at the same time. Service implementation (installation of X-Win32) and running appli-

cations on the Linux platform will provide a much richer computing environment, and adds support for legacy hardware and software. Interestingly, most of the applications (such as LyX, StarOffice, OpenOffice) are free of charge.

Chapter 5

ICOSYS2 Printing Services

5.1 Introduction

ICOSYS2 printing services allow users to send their print jobs to respective printers across the FTM network. Although FTM have allocated many printers to be shared among their staff, the printers are not properly managed to support network printing. This chapter describes the implementation of the ICOSYS2 network printing services. Section 5.2 outlines the service requirements. Section 5.3 details the implementation of the ICOSYS2 printing services. Section 5.4 summarizes the chapter.

5.2 Service Requirement

Printing is an important network service that must be made available in every computer networked environment. Printing services make networked printing peripherals and software easier to be used and managed. Currently, FTM printing is based on the Microsoft peer-to-peer printing environment. Printers are either locally attached to user's workstation or shared by a group of users.

The rudimentary arrangement is meant only for using the printers, or at least for providing a simple printer sharing facility. It is not meant to support the management

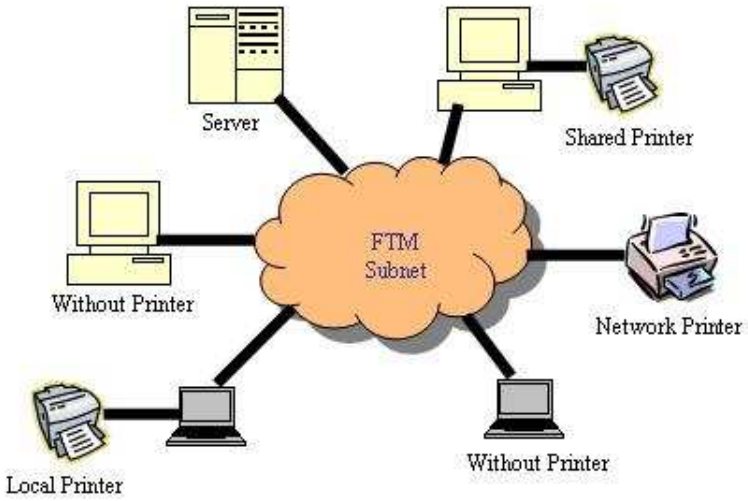


Figure 5.1: FTM Printing Resource Scenario

of printing resources. In other words, FTM does not have any centralized printing management which is important for enhancing printing capabilities among users and managing printing resources more effectively. Indeed, FTM is in dire need for network printing services.

The ICOSYS project attempts to introduce a proper solution for managing FTM network printing resources. In ICOSYS2, we propose and implement the open source Printing Management Services using CUPS (Common UNIX Printing Services). CUPS is very relevant for this purpose since the ICOSYS2 project is dedicated to culturalizing the open source effort in FTM. In addition, CUPS is a proven UNIX printing solution. CUPS is based on the Internet Printing Protocol (IPP), a new network printing protocol that enables us to print locally or remotely to networked printers. Any documents could be printed from Windows to open source environment and vice versa.

In ICOSYS2, the open source printing service is used to centralize all the print resources and fulfill users' and administrators' needs such as authentication, quota management, and printing job handling.

Figure 5.1 shows the printing resources scenario in the FTM subnet. By using

CUPS in the scenario depicted in Figure 5.1, the shared and network printers are linked together to the ICOSYS2 print server which functions as the printing manager. Users that require printing services will have to request through the print server. The print server will then direct the request to the selected printer which the user has been granted access. The ICOSYS2 print server also manages user authentication, user quotas, and printing jobs.

5.3 Service Implementation

This section describes the implementation of ICOSYS2 printing services using CUPS. There are two ways of installing CUPS, installing using binary source distribution or using RedHat Package Manager (RPM) distribution. In both cases, we have to consider the intallation requirements before starting the installation process. These will make sure there are no interruptions while installing CUPS.

5.3.1 Installing CUPS

5.3.1.1 Installing From Source Distribution

Installation Requirements

In order to build the CUPS system, the ANSI-compliant C and C++ compilers are needed. As its name implies, CUPS is designed to run on the UNIX operating system, however the CUPS interface library and most of the filters and backends supplied with CUPS should also compile and run under Microsoft Windows.

For the image file filters and PostScript RIP, JPEG, PNG, TIFF, and ZLIB, libraries are required. CUPS will build without these, but with significantly reduced functionality. Easy Software Products[1] maintains a mirror of the current versions of these libraries.

[1] ftp://ftp.easysw.com/pub/libraries

Compiling CUPS

CUPS uses GNU `autoconf` to configure the makefiles and source code for the system. Type the following command to configure CUPS into the system:

`./configure` ENTER

The default installation will put the CUPS software in the `/etc`, `/usr`, and `/var` directories onto the system, which will overwrite any existing printing commands on the system. By using the `--prefix` option to install the CUPS software in another location:

`./configure --prefix=/some/directory` ENTER

If the PNG, JPEG, TIFF, and ZLIB libraries are not installed in a system default location (typically `/usr/include` and `/usr/lib`), the CFLAGS, CXXFLAGS, and LDFLAGS environment variables prior to running configure need to be set:

`setenv CFLAGS "-I/some/directory"` ENTER

`setenv CXXFLAGS "-I/some/directory"` ENTER

`setenv LDFLAGS "-L/some/directory"` ENTER

`setenv DSOFLAGS "-L/some/directory"` ENTER

`./configure` ... ENTER or:

`CFLAGS="-I/some/directory";`

`export CFLAGS` ENTER

`CXXFLAGS="-I/some/directory";`

`export CXXFLAGS` ENTER

`LDFLAGS="-L/some/directory";`

`export LDFLAGS ENTER DSOFLAGS="-L/some/directory";`

`export DSOFLAGS` ENTER

`./configure` ... ENTER

Enabling support for encryption, the "`--enable-ssl`" option has to be added:

```
./configure --enable-ssl
```

SSL and TLS support require the OpenSSL library[2]. If the OpenSSL headers and libraries are not installed in the standard directories, the `--with-openssl-includes` and `--with-openssl-libs` options are needed:

```
./configure --enable-ssl \.
```

Once all the configuration has been made then start building the software.

5.3.1.2 Installing the Software From RPM Distribution

Besides installing from binary source distribution, it can also be installed from an RPM distribution. This is the simplest way to get CUPS ready to work.

5.3.2 Running CUPS

Once CUPS has been installed, start the CUPS server by typing: `/usr/sbin/cupsd` and then press ENTER.

5.3.3 Managing Printers

We use both KDEPrint and web-based utilities for the purpose of managing ICOSYS2 printing resources.

[2]available at: http://www.openssl.org

5.3.3.1 Web-based Printing Administration

The web interface is located at: `http://localhost:631/admin`. Administrators can perform all printing management tasks with few simple mouse clicks. Administrator's username and password (see Figure 5.2) are required to grant an access to CUPS server.

Figure 5.2: CUPS Web Password

If the authentication is valid, then the screen (see figure 5.3) is shown. The screen (namely Admin Starting Screen) consists of all the functions needed to manage CUPS printers such as adding and managing classes, managing jobs to be printed as well as adding and managing printers.

Figure 5.3: CUPS Web Admin

5.3.3.2 Creating Class

Before adding any printer, printer class needs to be created. Adding a class is simply by pressing the add class button. Fill in the information needed and then click continue (see Figure 5.4).

Figure 5.4: Add Class

Then Add SAMBA members to this class as needed. Succesfull class created is shown in Figure 5.5.

Figure 5.5: Successful Class Created

After creating a class, a printer can be added to the class. By selecting the add printer button in the class selection menu, a printer driver can be installed. Fill in the form with the information needed and then proceed to the next stages. In the devices selection menu select the Windows printer via SAMBA in the pull down textbox. After that choose a device URL. Fill in the printer device URL and then select the printer driver from the list. A successful printer added to the class is shown in figure 5.6.

Figure 5.6: Add Printer to the Class

5.3.3.3 Managing Printing Jobs

When a user requests to print via CUPS, a job list is created. Managing jobs is as simple as releasing a job and holding a job. Figure 5.7 shows how these tasks can be achieved.

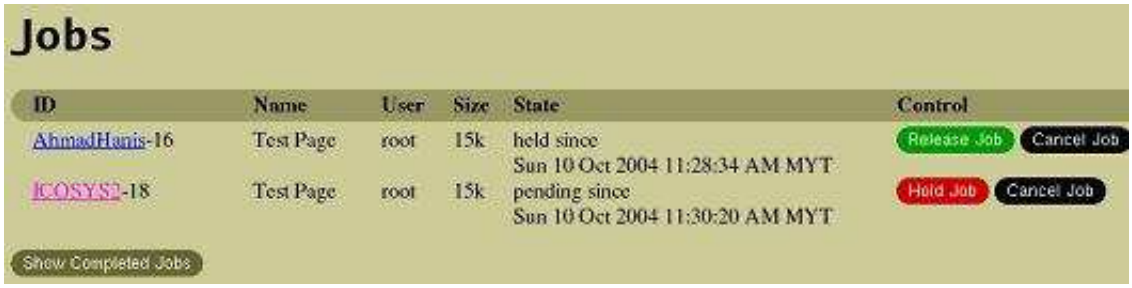


Figure 5.7: Jobs List

5.3.3.4 Setting Printing Quota

CUPS supports page and size-based quotas for each printer. The quotas are tracked individually for each user, but a single set of limits applies to all users for a particular printer. For example, we can limit every user to 5 pages per day on any printer, but we cannot limit each user.

The `job-k-limit`, `job-page-limit`, and `job-quota-period` options determine whether and how quotas are enforced for a printer. The `job-quota-period` option determines the time interval for quota tracking. The interval is expressed in seconds, so a day is 86,400, a week is 604,800 and a month is 2,592,000 seconds. The `job-k-limit` option specifies the job size limit in killobytes. The `job-page-limit` option specifies the number of pages limit.

For quotas to be enforced, the period and at least one of the limits must be set to a non-zero value. The following options will enable quotas:

```
/usr/sbin/lpadmin -p printer -o job-quota-period=
604800 -o job-k-limit=1024 ENTER
```

```
/usr/sbin/lpadmin -p printer -o job-quota-period=
604800 -o job-page-limit=100 ENTER
```

5.4 Summary

In this chapter, we have described our work in introducing a proper solution for managing FTM network printing resources using the ICOSYS2 printing services. For this purpose, we used CUPS to support the ICOSYS2 printing services - i.e. in managing shared and network printers on the FTM network.

CUPS is a printing system for UNIX and UNIX-like operating systems. This printing system enables us to print files to just about any printer we can find on the FTM network. CUPS has bridged the barrier between Windows and open source environments so that more reliable printing services can be handled on multi-platform environments. CUPS also provides printer drivers for many popular printers as well as advanced network printing management features including Web-based access to the printing systems, printer sharing, implicit classes, authentication, access control and encryption.

Chapter 6

ICOSYS2 Backup Services

6.1 Introduction

The ICOSYS2 project delivers the benefit of having a centralized file, and application storage and sharing environment for FTM citizens. Nevertheless, it is indeed not a wise decision for users to be fully dependent on the ICOSYS2 server to provide a secondary backup solution. The server itself is prone to failure. For this reason, we need a tertiary means to backup the user's data. This chapter describes the implementation of the ICOSYS2 backup services. Section 6.2 outlines the service requirements of the ICOSYS2 backup process. Section 6.3 describes the service implementation of the ICOSYS2 backup services. Section 6.4 summarizes the chapter.

6.2 Service Requirement

Since its establishment, FTM has no formal backup facilities for its citizens to backup their data. The current practice is that the users have to make their own backup either by using floppy disks or CDs. The problem exists because these disks cannot hold enough information to easily backup most modern computers. In addition, it is a burden for users to make their own backup for their data. Moreover, problems such as viruses,

burglaries and hard disk crashes will also be a great pain to the users who lose their data, especially when there is no means to get it back. Therefore, the ICOSYS2 project outlines a centralized backup facility to mitigate the risks and recover from data loss in the unlikely event of a system disaster.

Although ICOSYS2 already offers storage and file services, they might not be enough for the user. The fact that this service is a centralized-based system; if anything happens to the server, all the files and documents will perish. Also, if the user unintentionally deleted some important data in the server, the data is irretrievable. Thus, it is important for ICOSYS2 to implement (or at least, outline) a proper backup service to backup the server itself.

There are two means of backup; applications and data backup; ICOSYS2 concentrates on data backup since data is likely to be more important than applications. Normally, applications can be easily recovered.

6.3 Service Implementation

ICOSYS2 needs to offer backup facility services for a number of reasons. The two major reasons are; 1) a backup facility is very important to boost productivity, it saves time and energy, especially when we need to recover lost data, and 2) to increase data availability to be shared and created in a research activities environment. The ICOSYS2 team feels that the backup services will greatly help FTM by providing a proper way to maintain and protect important files (which include among others, exam questions, research data and reports) in the event of system failures, viruses and burglaries.

Nevertheless, due to the research's financial and timing constraints, the ICOSYS2 backup services cannot be performed by the ICOSYS2 research team on their own. The reason is that the tertiary backup requires additional hardware, software and human

resources (i.e., for maintaining backup and retrieval processes). The services will be provided by the University's Computer Center. Therefore, our work for this research will only outline the backup strategy on how the services can be implemented and provided to FTM citizens in the future.

6.3.1 Backup Strategy

As aforementioned, the implementation of ICOSYS2 backup services will be done off-site (i.e. performed remotely). The Computer Center will access the ICOSYS2 server in order to handle backup services remotely.

The implementation of the backup process will be based upon the **full backup strategy**. Under this strategy, the process will copy every file from user directories to a backup tape. Although it can be time consuming, is it beneficial due to quick restoration of files and the long term achievability. ICOSYS2 outlines that backup process for data files is performed on a weekly basis. This is based on our assumption that the user's data are not very regularly updated, hence reducing the need for daily backup.

The following backup strategy is suggested;

- Week 1 level 0 backup (full backup)
- Week 2 level 1 backup
- Week 3 level 1 backup
- Week 4 level 1 backup

During week 1, the process will perform a level 0 backup, which is a full directory backup. Then on weeks 2, 3 and 4, a level 1 backup will be performed. The level 1 backup is an incremental backup that only gathers all files that had been changed since the level 0 backup. This gives way to two basic backup and recovery strategies.

ICOSYS2 chose this kind of strategy because it requires only two sets of backup media. Restoring the full system from the level 0 backup and the previous week incremental can perform a complete restore.

The decision of this backup procedure is based on the capabilities of the Computer Center to provide the service. The incurred costs and responsibility involved in the provision of backup management is beyond the reach of FTM management. Backup is a continual effort and will incur cost to buy storages, compared to other services.

6.3.2 Backup Media

Since the ICOSYS2 backup services will be provided by the Computer Center, it is left to the Computer Center management to decide what backup technology (that includes backup software and hardware) will be used. Normally, magnetic tapes are used as backup media. This is due to the reason that the tape can handle up to 40 GB of data on a single reel and tape drives can write data at very high speeds. It is a low cost technology but comes with high storage capacities. Hence, tape is the commonly used media that allows a complete hard disk backup without requiring media swap during the process [12].

Standard tape media includes the DDS-4 with 20/40 GB capacity and maximum data transfer rate (DTR) of 2.4/4.8 Mbps. Figure 6.1 shows a commonly used backup medium.

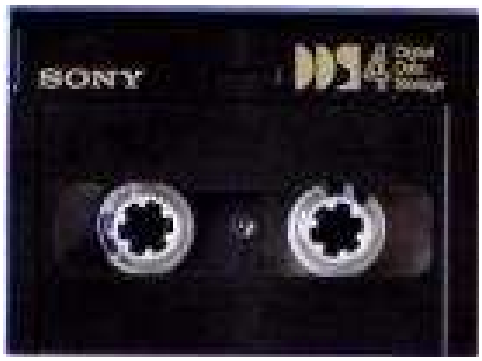


Figure 6.1: DDS-4 Tape Media

6.4 Summary

This chapter describes the ICOSYS2 backup services. In this chapter, we outlined a centralized backup facility to mitigate the risk of data loss and to recover lost data in the unlikely event of a system failure.

The implementation of the backup process will be based upon the **full backup strategy**, and is performed on a weekly basis. The ICOSYS2 services will be provided and supported by the University's Computer Center.

The ICOSYS2 backup services will greatly help FTM by providing a proper way of maintaining and protecting important files. The services require a continuous effort and recurring costs but will certainly be worthwhile in the long term.

Chapter 7

Conclusion

7.1 Research Contributions

This study was conducted to implement basic computer system services in FTM. These services include file and storage, application, printing and backup services. Theservices are important for FTM in order to fully utilize its computing facilities, so that it can enhance its image as an IT organization of the future.

The major contribution of this study is that it benefits FTM by providing basic server-based, centralized computing services to the academic community in FTM. Such services did not exist in FTM since the beginning. These services can now be utilized by FTM staff to facilitate their academic tasks such as research and teaching preparation.

As a summary, the contributions from the study are as follows:

- By completing this project, we have added more value to the University's investment on the provisions of the University's good network infrastructure. Prior to this, such a good LAN infrastructure has never been fully utilized. With ICOSYS2, we have introduced a new practice in FTM that will hopefully change the way we work and the way we use the network.

- We have outlined the baseline requirements for the provision of ICT services at an organization like FTM. These ICT services are common services that must be provided by such organizations.

- We have described an implementation of a basic file storage services that will provide FTM users with the capability of seamlessly sharing the file system across multiple system platforms. The provision of such service in ICOSYS2 offers better file protection for FTM users where they can save and backup their data on the network storage. Users can directly access their directories with the convenience of browsing their local system.

- We have implemented basic open source application services that equips FTM users with the capability to execute open source software without the need to install them locally on the users' machines.

- We have described and introduced a proper solution for managing FTM network printing resources using the ICOSYS2 printing services.

- We have outlined the requirements and the implementation strategies for having basic backup services in FTM.

- We have answered the call of our nation and its leaders in promoting the use of open source software in a way that is coherent with the National Open Source Policy.

7.2 Future Works

In summary, we consider that the implementation of ICOSYS2 project has successfully met its targetted objectives. Nevertheless, this project marks the beginning of our journey to change the working styles of the FTM citizens, in many aspects related

to the utilization of ICT. For this reason, several implementation issues need to be addressed in the future. These include:

- Strengthening the security aspects of this server-based environment.

  Several measures need to be undertaken to ensure the security of this environment. This includes a study on means to minimize the vulnerability of the environment from the external and internal security threats. Measures such as intrusion detection mechanisms need to be further explored.

- Enhancing interoperability between Windows and Linux -based systems.

  The interoperability issues among the heterogeneous systems that exist in the FTM working environment need to be studied. This includes researching on ways to manage heterogenous multi-platform systems environment especially on issues like multi-platform authentication services and cross platform interoperability issues.

- Measuring the performance of the ICOSYS2 environment.

  Performance studies in many aspects of the ICOSYS2 environment need to be measured and conducted in the future. This may include systems benchmarking and performance evaluation.

- Extending for future services and environment enhancement.

  Computing environments, like the ICOSYS, are dynamic in nature. New hardware, software, procedures and approaches needs to be considered in the future in the effort to enhance the ICOSYS2 environment.

- Policy development.

  We have introduced the ICOSYS2 systems services to the FTM citizens. What we have not done so far is the development of systems policies such as usage and security policies.

7.3 Our Final Words ...

We have accomplished what we envisioned almost twenty months ago - namely to design and implement an information and communication system services environment utilizing the open source approach on the FTM network.

We do hope that this research work will open another dimension in facilitating an efficient working environment in FTM. FTM staff can now use more open source products, instead of relying on the proprietary Microsoft-based applications and services. Many of these open source products are sufficiently powerful to support advanced computing research. In that respect, our move to use more open source products will surely strengthen our strategic competitiveness in the future.

Finally, this is just a humble beginning of our effort to cultivate the culture of working efficiently in an integrated state-of-the-industry computing environment. Please help us to do more in the future.

References

[1] Starnet Comm. Bringing UNIX/LINUX To Your Windows Desktop. Internet Article, 2003. http://www.starnet.com.

[2] F. Giunchiglia and T. Walsh. A Theory of Abstraction. *Artificial Intelligence*, 57(2-3):323–389, 1992.

[3] Microsoft. Microsoft Windows 2000 Test Lab. Internet Article, 2003. http://www.microsoft.com.

[4] Sun Microsystems. NFS: Network File System Protocol Specification. RFC 1094, March 1989. http://www.ietf.org.

[5] Mikalsen and Borgesen. *Local Area Network Management , Design and Security: A Practical Approach*. John Wiley, 2002.

[6] E. Nemeth, G. Snyder, S. Seebass, and T. Hein. *UNIX SYTEM administration Handbook*. Prentice Hall, 1995.

[7] K. Pfeifle. The KDEPrint Handbook. White Paper, 2001.

[8] R.World. Comparing Nine PC X Servers: Unix/Linux on Windows. Network Computer Magazine, June 2002.

[9] R. Sandberg. The Sun Network Filesystem: Design, Implementation, and Experience. In *Proceedings of the Summer 1986 USENIX Technical Confer-*

ence and Exhibition. USENIX Technical Conference and Exhibition, June 1986. http://citeseer.ist.psu.edu/sandberg86sun.html.

[10] M. Satyanarayanan, J. Kistler, P. Kumar, M. E. Okasaki, E. H. Siegel, and D. C. Steere. Coda: A Highly Available File System for a Distributed Workstation Environment. *IEEE Transactions on Computers*, 39(4):447–459, 1990. http://www.citeseer.ist.psu.edu/satyanarayanan90coda.html.

[11] M. Sweet. A Bright New Future for Printing on Linux. White Paper, 2004. http://www.linuxtoday.com.

[12] M. R. Sweet. *Common Unix Printing System*. Sams White Book. SAMS, first edition, August 2002.

[13] V. T. Toth. *Linux: A Low Cost Solution for Your Office*. SAMS, 1999.

[14] J. Vernooij, J. H. Terpstra, and G. Carter. The Official Samba-3 HOWTO and Reference Guide . Internet, November 2004. http://www.linuxjournal.com.