Unified Operating Systems

Ashish Patel, Vasundhara Misal, Pankaja Alappanavar, Ronak Agrawal

Abstract- Every Operating System has its different way of operation. When a novice user wants to perform some operations with the Operating System he is not acquainted with, the problem arises. The Novice user has to learn about the basic operations about the system to perform the intended task. However, this is time consuming job and often leads to frustration when needs to be done frequently. Hence, leads to reduced productivity. One answer to the above mentioned problem can be a generic interface which would allow user to perform his task irrespective of the underlying Operating System. Under these circumstances this paper proposes a system which implements the above mentioned interface as the core concern. The unique feature that the above implemented system will provide is the same input and output syntax for performing the intended tasks under the scope of the system.

Studied statistics show that this system is capable of achieving an Operating System independent interface on all JAVA supported systems.

Keywords: Operating System, Working Platform, Java Swing, GUI.

I. INTRODUCTION

In Industry, many times a developer has to switch between the underlying Operating Systems. Each operating system has its own functionalities. If a developer is a novice user then he has to learn the new functionalities of underlying operating system. But it is not possible to learn different functionalities and different commands of different operating system. Also the user can be master of the single operating system but not more than one and no user is interested to be master of more than one operating system. So the solution is *unified operating system. Unified operating system* has generic interface which provides novice user to work with ease in any underlying operating system. in fact helps novice user from realizing that he has switched in between work environment, keeping   the performance and the speed of a user unaffected due to switch.

   Mr. Ashish Patel, Department Information Technology**,** Sinhgad Academy of Engineering, Pune, India. 9096667265 (Email: meetashishk.patel@gmail.com).

   Miss. Vasundhara Misal, Department Information Technology**,** Sinhgad Academy of Engineering, Pune, India. 9527497060 (Email: vasu9119@gmail.com).

   Mrs. Pankaja Alappanavar, Department of Information Technology**,** Sinhgad Academy of Engineering, Pune, India. 9881376586 (Email: pankaja.ba@gmail.com).

   Mr. Ronak Agrawal, Department Information Technology**,** Sinhgad Academy of Engineering, Pune, India. 9028112207 (Email: ronakagrawal2006@gmail.com).

II. EXISTING SYSTEMS

Existing Technologies include two leading tools by market biggies. It includes

A. VMware Workstation:

• It is desktop software. It allows you to run multiple x86-based desktop and server operating systems simultaneously on a single PC. With VMware Workstation, you spend less time procuring and configuring, and more time testing and deploying. Over three million software development, quality assurance, and IT professionals worldwide find VMware Workstation an indispensable tool. Some benefits of VMware workstation: Run multiple operating systems simultaneously on a single physical machine

• Revolutionize software development and testing
• Enhance productivity of enterprise IT professionals
• Facilitate team collaboration
• Introduce Virtual Infrastructure to your enterprise

Limitations of VMware workstation:
• Guest system maximum RAM: 64 GB
• Number of guest CPUs: 4
• Number of hosts in an HA cluster: 32
• Number of hosts in a DRS cluster: 32
• Size of RAM per server: 256 GB
• Number of hosts managed by Virtual Center Server: 200
• Number of virtual machines managed by Virtual Center Server: 2000

B. Oracle VM Virtual Box:

It installs on your existing computer, whether they are running Windows, Mac, Linux or Solaris operating system. It extends the capabilities of your existing computer so that it can run multiple operating systems at the same time. Some benefits of VM Virtual box:
• Running multiple operating systems simultaneously
• Easier software installations
• Testing and disaster recovery
• Infrastructure consolidation
•

Limitations of VMware workstation:
• Poor performance
• 64-bit guests on some 32-bit host systems with VT-**x** can cause instabilities
Compacting virtual disk images
is limited to VDI file

III. SCOPE OF UNIFIED OPERTING SYSTEM

The *Unified Operating System* has the same input and output syntax irrespective of the underlying operating system. Also it will provide user with an easy to learn set of commands will allow successful implementation of the concept of unifying operating systems.
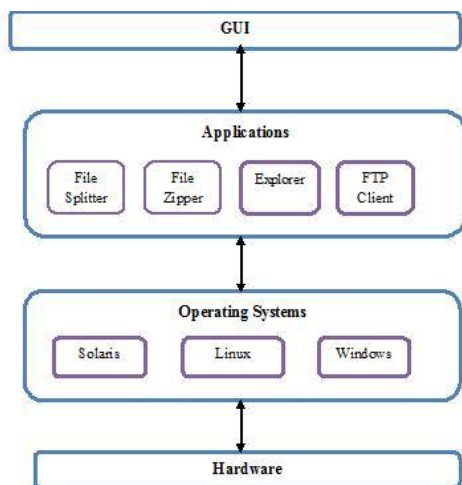

Fig.1 Block diagram of Unified Operating System

This system will be helpful for the users working on an altogether different operating system. It will also provide user with a set of advanced utility operations including but not limited to the following:

A. File Zipper

It will provide with management of archive files. Zipping file is one of the basic functionality where every user uses this task to compress their data. Recursion is used for iterating the directories. The normal compression ratio is about 1:10. Here the folder and single file can also be zipped.

Zip is a file format used for data compression and archiving. A zip file contains one or more files that have been compressed, to reduce file size. The zip file format permits a number of compression algorithms [3].

Zip is used as a base file format by many programs, usually under a different name. Zip files are often represented by a document or other object prominently featuring a zipper.

A zip file is identified by the presence of a *central directory* as shown in Fig. 2 is located at the end of the structure in order to allow the appending of new files. The central directory stores a list of the names of the entries of files or directories stored in the zip file, along with other metadata about the entry, and an offset into the zip file, pointing to the actual entry data. This allows a file listing of the archive to be performed relatively quickly, as the entire archive does not have to be read to see the list of files. The entries in the zip file also include this information for redundancy.

The order of the file entries in the directory need not coincide with the order of file entries in the archive.
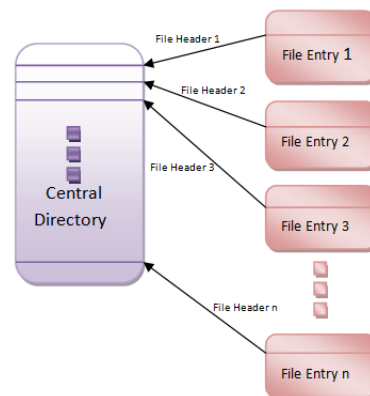

Fig.2 Structure of Zip File

Each entry is introduced by a local header with information about the file such as the comment, file size and file name, followed by optional "Extra" data fields, and then the possibly compressed, possibly encrypted file data. The "Extra" data fields are the key to the extensibility of the zip format. "Extra" fields are exploited to support the ZIP64 format, WinZip-compatible AES encryption, file attributes, and UNIX file timestamps. Other extensions are possible via the "Extra" field. Zip tools are required by the specification to ignore Extra fields they do not recognize.

There is no BOF or EOF marker in the zip specification. Often the first thing in a zip file is a zip entry, which can be identified easily by its signature. But it is not necessarily the case that a zip file begins with a zip entry, and is not required by the zip specification.

B. FTP Client

File Transfer Protocol (FTP) is a standard network protocol used to transfer files from one host to another host over a TCP -based network, such as the internet. It is often used to upload web pages and other documents from a private development machine to a public web-hosting server. FTP is built on client-server architecture and uses separate control and data connections between the client and the server. FTP users may authenticate themselves using a clear-text sign-in protocol, normally in the form of a username and password, but can connect anonymously if the server is configured to allow it.

FTP may run in *active* or *passive* mode, which determines how the data connection is established. In active mode, the client sends the server an IP address and port number and then waits until the server initiates the TCP connection. In situations where the client is behind a firewall and unable to accept incoming TCP connections, *passive mode* may be used. In this mode, the client sends a PASV command to the server and receives an IP address and port number from the server, which the client then uses to open a data connection to the server [4].

While transferring data over the network, four data representations can be used:

- ASCII mode: used for text. Data is converted, if needed, from the sending host's character representation to"8-bit ASCII"before transmission, and again, if necessary to the receiving host's character representation. As a consequence, this mode is inappropriate for files that contain data other than plain text.
- Image mode: the sending machine sends each file byte for byte, and the recipient stores the byte stream as it receives it.
- EBCDIC mode: use for plain text between hosts using the EBCDIC character set. This mode is otherwise like ASCII mode.
- Local mode: Allows two computers with identical setups to send data in a proprietary format without the need to convert it to ASCII

Login

Fig 3 shows FTP login procedure. It utilizes a normal username and password scheme for granting access. The username is sent to the server using the USER command, and the password is sent using the PASS command. If the information provided by the client is accepted by the server, the server will send a greeting to the client and the session will commence. If the server supports it, users may log in without providing login credentials, but the server may authorize only limited access for such sessions.
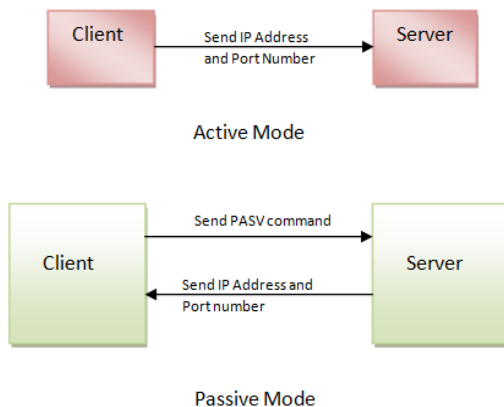

Fig.3 FTP Login procedure

C.   File Explorer

File Explorer is the core functionality that every uses. Nothing is more frustrating than struggling with the File System at disposal. File Explorer displays the hierarchical structure of _les, folders, and drives on your computer. This functionality will deal with the basic functions of Cut, Copy, Paste, Delete and Rename. All the basic file operations will be provided at one go irrespective of the operating system under implementation. All the above functions will be mapped accordingly to perform the intended operation[4].

D.   File Splitter

This functionality implements a tool where which will split large file into small chunks so that large file can be transferred easily via network or in the storage devices. File splitter takes the help of _le encoder for buffering when the file is split as well as naming conventions of the split files.

A demultiplexer for digital media files, or media demultiplexer also called a file splitter by laymen or consumer software providers, is software that demultiplexes individual elementary of a media file, *e.g.*, audio, video, or subtitles and sends them to their respective decoders for actual decoding. Media demultiplexers are not decoders themselves, but are format container handlers that separate program streams from a file and supply them to their respective audio, video, or subtitles decoders[4].

IV. SYSTEM Analysis

The system we will develop will be taking into account Red Hat Enterprise Linux and Solaris Operating Systems. The selection of the above operating system is due to the following characteristics possessed by each of them.

- VM is configured best on Solaris. Also Performance tuning at optimum level can be achieved in Solaris.
- JAVA performs best on Solaris.
- It implements the latest File System "Zettabyte" File System.
- Red Hat on the other hand has Generic Boot Loader. Thus, provides maximum flexibility.
  Key features provided by system would include:
- Developing a Platform Independent Application.
- Providing user with a Hassle-free Interface.
- Implementation with a Strong and user-friendly GUI.
- Same Input and Output Syntax for implemented Operating Systems.

V. APPLICATIONS

This system would be used Solaris and Red Hat Enterprise Linux with a common interface and provide user with a user friendly and known environment, without being concerned about the OS the user is working with.

In the future works, Unified Operating System could be a crucial tool for moving up from one system to another. We need to consider the way in which we could combine the power of various operating systems into single unit without actually disturbing the kernel.

VI. CONCLUSION

We have discussed a scheme for implementation of a unified interface which would provide user with a common interface for all the basic operations irrespective of the underlying Operating System. This system could provide us with the power to combine the strength of different Operating Systems into one single entity. For current system under design, the scope is limited to the implementation of core functionalities. This system greatly reduces the overhead faced by a novice user when getting acquainted with a new work environment.

REFERENCES

[1]. Moving from Solaris to Red Hat Enterprise Linux | Michael Solberg
[2]. Forouzan, B.A. (2000). TCP/IP: Protocol Suite. 1st ed. New Delhi, India: Tata McGraw-Hill Publishing Company Limited.
[3]. APPNOTE.TXT - .ZIP File Format Specification, Version: 4.5, 2001-11-01, retrieved 2012-01-05
[4]. Unifying Operating System published in NTSACT-2012 (ISBN 978-1-4675-1444-6)