Proceedings of National Conference on Recent Advances in Electronics and Communication Engineering (RACE-2014), 28-29 March 2014

Comparison of different Operating System

Niti gupta [1], Amrita ticku[2], Manoj kumar[3]
[1,2]Faculty: Department of cse at Dronacharya Group Of Institution
Knowledge Park-III, Greater Noida, Uttar Pradesh, India
[3]Student: Department of cse at Dronacharya Group Of Institution
Knowledge Park-III, Greater Noida, Uttar Pradesh, India
[1] nitigupta86@gmail.com, [2]amrita_koul27@gmail.com, [3]mnj_gpt@rediffmail.com.

Abstract-Operating System: In the current era the OS is used in every Mobile, Laptop, Tablets and Desktops. Day by day there is some important in Operating System and every new development give birth to a new Technology and new Operating System. People's requirements are changing as the time change. They want to keep themselves update. Our Paper just gives a refreshing review on Operating System that has been developed.
This paper will help to compare operating Systems by their technology and usage in all aspect so that everyone can choose best according to their requirements.

I. INTRODUCTION

Operating system (**OS**) is a collection of software that manages computer hardware resources and provides common services for computer programs. The operating system is an essential component of the system software in a computer system. Application programs usually require an operating system to function. Operating System can be defined as "A program that acts as an intermediary between a user of a computer and the computer hardware" Goals of Operating System Are:

- ✓ Execute user programs and make solving user problems easier
- ✓ Make the computer system convenient to use
- ✓ Use the computer hardware in an efficient manner.

Operating systems can be found on almost any device that contains a computer from cellular phones and video game consoles to supercomputers and web servers.

I.I ARCHITECTURE OF OPERATING SYSTEM

Kernel Mode
In Kernel mode, the executing code has complete and unrestricted access to the underlying hardware. It can execute any CPU instruction and reference any memory address. Kernel mode is generally reserved for the lowest-level, most trusted functions of the operating system.

User Mode
In User mode, the executing code has no ability to directly accesshardware or reference memory. Code running in user mode must delegate to system APIs to access Hardware or memory. As shown in figure:-1.

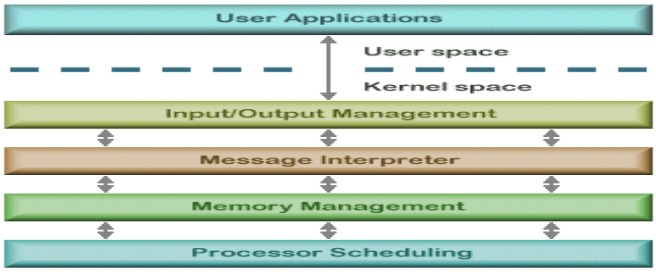

Figure.1

Services provided by the Operating System

- **User Interfaces -** Means by which users can issue commands to the system. Depending on the system these may be a command-line interface (e.g. sh, csh, ksh, tcsh, etc.), a GUI interface (e.g. Windows, X-Windows, KDE, Gnome, etc.), or a batch command systems. The latter are generally older systems using punch cards of job-control language, JCL, but may still be used today for specialty systems designed for a single purpose.
- **Program Execution -** The OS must be able to load a program into RAM, run the program, and terminate the program, either normally or abnormally.
- **I/O Operations -** The OS is responsible for transferring data to and from I/O devices, including keyboards, terminals, printers, and storage devices.
- **File-System Manipulation -** In addition to raw data storage, the OS is also responsible for maintaining directory and subdirectory structures, mapping file names to specific blocks of data storage, and providing tools for navigating and utilizing the file system.
- **Communications -** Inter-process communications, IPC, either between processes running on the same processor, or between processes running on separate processors or separate machines. May be implemented as either shared memory or message passing, (or some systems may offer both.)
- **Error Detection -** Both hardware and software errors must be detected and handled appropriately, with a minimum of harmful repercussions. Some systems may include complex error avoidance or recovery systems, including backups, RAID drives, and other redundant systems. Debugging and diagnostic tools aid users and administrators in tracing down the cause of problems.

Other systems aid in the efficient operation of the OS:

- **Resource Allocation -** E.g. CPU cycles, main memory, storage space, and peripheral devices. Some resources are managed with generic systems and others with very carefully designed and specially tuned systems, customized for a particular resource and operating environment.
- **Accounting -** Keeping track of system activity and resource usage, either for billing purposes or for statistical record keeping that can be used to optimize future performance.
- **Protection and Security -** Preventing harm to the system and to resources, either through wayward internal processes or malicious outsiders. Authentication, ownership, and restricted access are obvious parts of this system. Highly secure systems may log all process activity down to excruciating detail, and security

regulation dictate the storage of those records on permanent non-erasable medium for extended times in secure (off-site) facilities.

- **System calls-** provide a means for user or application programs to call upon the services of the operating system.Generally written in C or C++, although some are written in assembly for optimal performance.

I.II COMPONENTS OF OPERATING SYSTEM:

- The components of an operating system (as shown in figure:-2) all exist in order to make the different parts of a computer work together. All user software needs to go through the operating system in order to use any of the hardware, whether it be as simple as a mouse or keyboard or as complex as an Internet component.

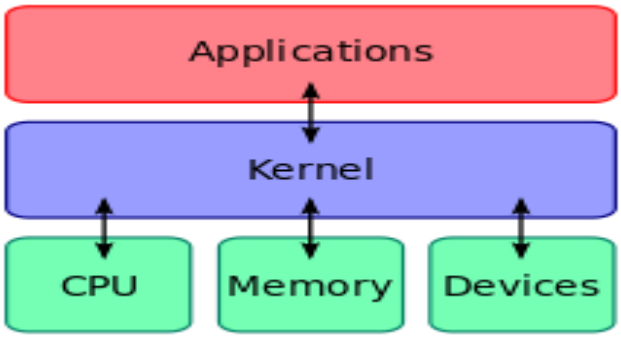

Figure-2 Component of Operating System

Kernel

With the aid of the firmware and device drivers, the kernel provides the most basic level of control over all of the computer's hardware devices. It manages memory access for programs in the RAM, it determines which programs get access to which hardware resources, it sets up or resets the CPU's operating states for optimal operation at all times, and it organizes the data for long-term non-volatile storage with file systems on such media as disks, tapes, flash memory, etc.

Program Execution

The operating system provides an interface between an application program and the computer hardware, so that an application program can interact with the hardware only by obeying rules and procedures programmed into the operating system. The operating system is also a set of services which simplify development and execution of application programs. Executing an application program involves the creation of a process by the operating system kernel which assigns memory space and other resources, establishes a priority for the process in multi-tasking systems, and loads program binary code into memory, and initiates execution of the application program which then interacts with the user and with hardware devices.

Interrupts

Interrupts are central to operating systems, as they provide an efficient way for the operating system to interact with and react to its environment. The alternative — having the operating system "watch" the various sources of input for events (polling) that require action — can be found in older systems with very small stacks (50 or 60 bytes) but are unusual in modern systems with large stacks. Interrupt-based programming is directly supported by most modern

CPUs. Interrupts provide a computer with a way of automatically saving local register contexts, and running specific code in response to events. Even very basic computers support hardware interrupts, and allow the programmer to specify code which may be run when that event takes place. When an interrupt is received, the computer's hardware automatically suspends whatever program is currently running, saves its status, and runs computer code previously associated with the interrupt; this is analogous to placing a bookmark in a book in response to a phone call. In modern operating systems, interrupts are handled by the operating system's kernel. Interrupts may come from either the computer's hardware or from the running program.

When a hardware device triggers an interrupt, the operating system's kernel decides how to deal with this event, generally by running some processing code. The amount of code being run depends on the priority of the interrupt (for example: a person usually responds to a smoke detector alarm before answering the phone). The processing of hardware interrupts is a task that is usually delegated to software called device driver, which may be either part of the operating system's kernel, part of another program, or both.

I.III TYPES OF OPERATING SYSTEM

Real-time Operating System

A real-time operating system is a multitasking operating system that aims at executing real-time applications. Real-time operating systems often use specialized scheduling algorithms so that they can achieve a deterministic nature of behaviour. The main objective of real-time operating systems is their quick and predictable response to events. They have an event-driven or time-sharing design and often aspects of both. An event-driven system switches between tasks based on their priorities or external events while time-sharing operating systems switch tasks based on clock interrupts.

Multi-user Operating System

A multi-user operating system allows multiple users to access a computer system at the same time. Time-sharing systems and Internet servers can be classified as multi-user systems as they enable multiple-user access to a computer through the sharing of time. Single-user operating systems have only one user but may allow multiple programs to run at the same time.

Multi-tasking Operating System

A multi-tasking operating system allows more than one program to be running at the same time, from the point of view of human time scales. A single-tasking system has only one running program. Multi-tasking can be of two types: pre-emptive and co-operative. In pre-emptive multitasking, the operating system slices the CPU time and dedicates one slot to each of the programs. Unix-like operating systems such as Solaris and Linux support pre-emptive multitasking, as does AmigaOS. Cooperative multitasking is achieved by relying on each process to give time to the other processes in a defined manner. 16-bit versions of Microsoft Windows used cooperative multi-tasking. 32-bit versions of both Windows NT and Win9x used pre-emptive multi-tasking. Mac OS prior to OS X used to support cooperative multitasking.

Distributed Operating System

A distributed operating system manages a group of independent computers and makes them appear to be a single computer. The development of networked computers that could be linked and communicate with each other gave

rise to distributed computing. Distributed computations are carried out on more than one machine. When computers in a group work in cooperation, they make a distributed system.

Embedded Operating System

Embedded operating systems are designed to be used in embedded computer systems. They are designed to operate on small machines like PDAs with less autonomy. They are able to operate with a limited number of resources. They are very compact and extremely efficient by design.

Time Sharing Operating System

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

Examples of popular modern operating systems include Android, BSD, iOS, Linux, OS X, QNX, Microsoft Windows, Windows Phone, and IBM z/OS. All these, except Windows, Windows Phone and z/OS, share roots in UNIX.

II COMPARISON OF OPERATING SYSTEM

The following table provide the comparison between General and Technical information for a number of widely used and currently available PC and handheld (including smart phone and tablet computer) operating systems. Thepaper share the information of operating systems provides a broader, and more general, comparison of operating systems that includes servers, mainframes and supercomputers. There are large numbers of variety of Linux Operating System. See the comparison of Linux distributions for a detailed comparison. There are also a variety of BSD operating systems, covered in comparison of BSD operating systems.(as shown in table :-1).

OS Name	**Latest stable version**	**Latest release date**	**Target system type**	**OS Name**	**Latest stable version**	**Latest release date**	**Target system type**
AIX f	7.1	2010	Server, NAS, workstation	**DragonFly BSD**	3.6	2013	Server, workstation, NAS, embedded
Android	4.4 (KitKat)	2013, October 31	Consumer, enterprise, military, education	**Haiku**	R1/Alpha4	2012	Personal computer

Amiga OS classic	3.9	2000, December 4	Workstation, personal computer	**HP-UX**	11.31 "11i v3"	2007	Server, workstation
Amiga OS 4	4.1 update 6	2012	Workstation, personal computer	**IBM i**	7.1	2010	Server
eComStation	2.1	2011	Server, workstation, personal computer	**Inferno**	Fourth Edition	2007	NAS, server, embedded
EPOC32	ER5	1999	PDA	**iOS**	7.0	2013	Smartphone, music player,tablet computer
FreeBSD	9.2	2013	Server, workstation, NAS,embedded	**IRIX**	6.5.30	2006	Server, workstation
GhostBSD	3.1	2013	Desktop,	**Linux**	3.12.2	2013, Nov	Comparision of linux
DragonFly BSD	3.6	2013	Server, workstation, NAS, embedded	**MPE**	MPE-V	1988	Server
Haiku	R1/Alpha4	2012	Personal computer	**MPE/XL**	7.5	2002	Server
	Latest	**Latest**	**Target**		**Latest**	**Latest**	**Target**

OS Name	stable version	release date	system type	OS Name	stable version	release date	system type
IBM i	7.1	2010	Server	**OS X Server**	10.9	2013	Server
Inferno	Fourth Edition	2007	NAS, server, embedded	**MINIX 3**	3.2.1	2013	Workstation
iOS	7.0	2013	Smartphone, music player,tablet computer	**NetBSD**	6.1.1	2013	NAS, server, workstation, embedded
IRIX	6.5.30	2006	Server, workstation	**NetWare**	6.5 SP8	2008	Server
Linux	3.12.2	2013, November 29	See: Comparison of Linux distributions	**NeXTStep**	3.3	1995	Workstation
NetBSD	6.1.1	2013	NAS, server, workstation, embedded	**OpenIndiana**		2010, December 17	Server, workstation
NetWare	6.5 SP8	2008	Server	**OpenVMS**	8.4	2010	Server, workstation
OpenIndiana		2010,	Server,	**OpenVMS**	8.4	2010	Server, workstation
OpenIndiana		2010, December 17	Server, workstation	**RISC OS**	4.39	2004	Education,
OpenVMS	8.4	2010	Server, workstation	**RISC OS**	5.18	2012	Education, personal computer

OS Name	Latest stable version	Latest release date	Target system type	OS Name	Latest stable version	Latest release date	Target system type
PC-BSD	9.2	2013	Personal computer, workstation, server	**ZETA**	1.5	2007	Personal computer, mediacenter, workstation
Plan 9	Fourth Edition	(Daily snapshots)	Workstation, server, embedded, HPC	**STOP 6, XTS-400**	6.4.U1	2007	Server, workstation
QNX	6.5.0	2012	Workstation, smartphones, consumer, server, industrial, automotive, embedded	**ReactOS**	0.3.15	2013	Workstation, personal computer
Solaris	11.1	2012	Server, workstation	**VxWorks**	6.9	2011	Embedded Real-time systems
Symbian	9.5	2009	Phones	**z/OS**	1.12	2010	IBM mainframe
Symbian platform	3.0.4	2010	Embedded	**z/VSE**	5.1.1	2012, June	IBM mainframe
Windows Server (NT family)	Windows Server 2012 R2 (NT 6.3.9600)	2012	Server, NAS, embedded, HPC	**z/VM**	6.3	2013	IBM mainframe
Microsoft Windows(NT family)	Windows 8.1(NT 6.3.9600)	2013	Workstation, personal computer, media center,Tablet PC, embedded	**HP Nonstop**	H06.24/J06.13	2012	HP Nonstop Servers

TABLE:-1

III TECHNICAL INFORMATION

1. Operating systems where the GUI is not installed and turned on by default are often bundled with an implementation of the X Window System, installation of which is usually optional.

2. Most operating systems use proprietary APIs in addition to any supported standards.

3. Amiga OS features since OS 2.0 version a standard centralized Install utility called Installer, which could be used by any software house to install programs. It works as a Lisp language interpreter, and install procedures could be listed as simple text. AmigaOS can also benefit of a 3rd party copyrighted library called XAD that is available for all POSIX (Unix, Linux, BSD, and for AmigaOS, MorphOS, etc.). This library is freely distributable and publicly available on Aminet Amiga centralized repository of all Open Source or Free programs and utilities. XAD.Library, complete with GUI Voodoo-X, is based on modules and capable to manage over 300 compression methods and package systems (Voodoo-X GUI supports 80 package systems), including those widely accepted as standards such as .ZIP, .CAB, .LHA, .LZX, .RPM, etc.

4. A standard AmigaOS installation requires usually only few files (typically 3 to 10 files) to be copied in their appropriate directory, and libraries and language files for national localization to be put in their standard OS directories. Any Amiga user with some minimal experience knows where these files should be copied and could perform programs installations by hand.

5. AmiUpdate is capable to update AmigaOS files and also all Amiga programs which are registered to use the same update program that is standard for Amiga. Updating AmigaOS requires only few libraries to be put in standard OS location (for example all libraries are stored in "Libs:" standard virtual device and absolute path finder for "Libs" directory, Fonts are all in "Fonts:" absolute locator, the files for language localization are all stored in "Locale:" and so on). This leaves Amiga users with a minimal knowledge of the system almost free to perform by hand the update of the system files.

6. NetBSD and OpenBSD include the X Window System as base install sets, managed in their respective main source repository, including local modifications. Packages are also provided for more up-to-date versions which may be less tested.

7. Windows can read and write with Ext2 and Ext3 file systems only when a driver from FS-driver or Ext2Fsd is installed. However, using Explore2fs, Windows can read from, but not write to, Ext2 and Ext3 file systems. Windows can also access ReiserFS through rfstool and related programs.

2. IV Securities

(Refer as shown in table 2).

Name	**Resource access control**	**Integrated firewall**	**Encrypted file systems**	**Name**	**Resource access control**	**Integrated firewall**	**Encrypted file systems**
AIX 7.1	POSIX,ACLs,MAC,Trusted AIX - MLS,RBAC	IPFilter,IPsecVPNs, basic IDS	Yes	**OS X 10.6.4**	POSIX, ACLs[s 6]	ipfw	Yes

Name	Resource access control	Integrated firewall	Encrypted file systems	Name	Resource access control	Integrated firewall	Encrypted file systems
GhostBSD 3.1	POSIX, ACLs, MAC	IPFW2, IPFilter, PF	Yes	**NetWare 6.5 SP8**	Directory-enabled ACLs	IPFLT.NLM	Yes
HP-UX	POSIX, ACLs	IPFilter	No	**OES-Linux**	Directory-enabled ACLs	IPFilter	Yes
Inferno	POSIX	***?***	***?***	**OpenBSD 4.8**	POSIX	PF	Yes
Name	**Resource access control**	**Integrated firewall**	**Encrypted file systems**	**Name**	**Resource access control**	**Integrated firewall**	**Encrypted file systems**
Linux 2.6.39	POSIX, ACLs,[s 4]MAC	Netfilter, varied by distribution	Yes	**OpenVMS 8.4**	ACLs,privileges	***?***	***?***
Mac OS 9.2.2	No	No	No	**OS/2,eComStation**	ACLs[s 7]	IPFilter	No
Plan 9	POSIX ?	ipmux	Yes	**OpenSolaris2009.06**	POSIX, RBAC, ACLs, least privilege, Trusted Extensions	IPFilter	Yes[s 10]
QNX 6.5.0	POSIX	PF, from NetBSD	***?***	**Windows Server 2008R2**	ACLs, privileges, RBAC	Windows Firewall	Yes
RISC OS	No	No	No	**Windows 7SP1**	ACLs, privileges, RBAC	Windows Firewall	Yes
Solaris 10	POSIX, RBAC, ACLs,least privilege,Trusted Extensions	IPFilter	Yes[s 10]	**Windows Vista SP2**	ACLs, privileges, RBAC	Windows Firewall	Yes
STOP 6, XTS-400[s 14]	POSIX,multilevel security,Biba Modelmandatory integrity, ACLs, privileges, subtype mechanism	No	No	**Windows XPPro SP3**	ACLs	Windows Firewall	Yes, with NTFS
z/OS 1.11	RACF	z/OS IPSecurity	Optional	**ZETA**	POSIX[s 13]	No	No

TABLE:-2

V CONCLUSIONS

We conclude that among **Windows family** Windows 7 is fast , easy to understand , simple and has excellent user experience.It is user-friendly and explorer friendly whereas Windows 8 is a total different thing that has been created. Windows 7, xp and 98 are somewhat same but windows8 is not even a bit same as them. And Windows 7 is also the most popular in the croud of a lot of people who love computers but Windows 8 has more security features and adds an extra twist and extra compatibility to Windows.

Moving towards other operating systems, **Ubuntu** is a mixture of Windows and Macintosh. It looks like mac but shortcut keys is similar to Windows**. Macintosh** is one of the easiest operating system to learn for a complete beginner (although switching from windows has a slight learning curve). It is ideal for the everyday user but Linux is better than Ubuntu**. Linux** is very similar to other operating systems, such as Windows and OS X . Linux is already successful on many different kinds of devices, but there are also many technological areas where Linux is moving towards, even as desktop and server development continues to grow faster than any other operating system today. Now there are mobile operating system also such as **Symbian** and **Android**.

Symbian is a mobile operating system (OS) targeted at mobile phones that offers a high-level of integration with communication and personal information management (PIM) functionality. Symbian OS combines middleware with wireless communications through an integrated mailbox and the integration of Java and PIM functionality (agenda and contacts) on the other hand, Android is the operating system that powers over 1 billion smartphones and tablets. Since these devices make our lives so sweet, each Android version is named after a dessert: Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwich, and Jelly Bean ,Kitkat. Android is better than Symbian.

Now if talk about **Solaris** and **Open solaris**, **Solaris** is a Unix operating system originally developed by Sun Microsystems. It superseded their earlier SunOS in 1993. **Oracle Solaris**, as it is now known, has been owned by Oracle Corporation since Oracle's acquisition of Sun in January 2010.Solaris is known for its scalability on the other hand open solaris is an open source operating system, similar in scope to GNU/Linux and BSD, but descended from the proprietary Solaris operating system from Sun Microsystems. Therefore solaris is better than open solaris.

There are operating systems that are based on Unix OS.These are ZETA , HP-UX, BSD.**HP-UX(Hewlett-PackardUniX)** is Hewlett-Packard's proprietary implementation of the Unix operating system, based on UNIX System V (initially System III) and first released in 1984.**magnussoft ZETA**, earlier **yellowTAB ZETA**, was an operating system formerly develop.

VI REFERENCES

[1] Operating system - Wikipedia, the free encyclopedia en.wikipedia.org/wiki/Operating_system

[2] How Stuff Works "Types of Operating Systems"www.howstuffworks.com/operating-system3.htm

[3] Computer Operating Systems - Computer Hope www.computerhope.com/os.htm

[4] Operating System Types - Tutorials Point www.tutorialspoint.com/operating_system/

[5] Operating Systems http://linux.about.com/od/glossary/a/glsopsystems.htm

[6] What is a Operating System? http://www.ask.com/question/microkernel-operating- system.

[7] Operating System - An IT Definition FromWWW. Webopedia.com

[8] HowStuffWorks "Types of Operating Systems"http://www.howstuffworks.com/operating-system3.htm

[9] Information on different mobile Operating Systems http://mobiledevices.about.com/od/operatingsystem

[10] What Are the Different Types of Operating Systemhttp://www.ask.com/question/what-are-the-different-type.

[11] Computer Operating Systems - Computer Hope http://www.computerhope.com/os.htm

[12] Teach ICT - GCSE ICT - types of operating systems, http://www.teachict.com/gcse_new/computer%20system

[13] What's 10 different types of operating system? http://answers.yahoo.com/question/index

[14] A Comparative Study of Mobile Phone's Operatin Systems - ijcaitwww.ijcait.com/IJCAIT/index.php

[15] CS261: Research Topics in Operating Systems (Fall 2013)www.eecs.harvard.edu/cs261/

[16] research.microsoft.com/en-us/groups/os/

[17] www.clarkson.edu/~jnm/cs644