

Contents:

1 - INTRODUCTION:..... 2

2- BACKGROUND INFO 3

3 - ANALYSIS:..... 6

 A - NECESSITY FOR AN OPERATING SYSTEM: 6

 B - TECHNOLOGICAL DEVELOPMENT AND EVOLUTION: 7

 C - EXISTING STANDARDS: 21

 D - TIMELINE:..... 23

4 - CONCLUSION:..... 30

5 - APPENDICES: 31

 FIGURES:..... 31

 TABLES:..... 35

6 - REFERENCES:..... 36

1 - Introduction:

New computer technologies have changed many different aspects of human lives. These technologies are in fact, the products of long and very advanced information system researches. The output of these researches, which have been practiced since the Second World War, opened new and exciting ways of realizing some of our dreams and managing our businesses. Advanced computer standards can be seen as one of the important outputs of these productive advances in the computer technology field. This type of technology brought many changes and new improvements. As a result, computers became part of daily life. They are widely used all around the world to communicate, to create, and to profit.

One of the mentioned improvements is the creation of operating systems. An operating system that will be deeply defined later in this part of the paper is the software, controlling the allocation and usage of hardware resources such as memory, CPU time, disk space, and peripheral devices. “In other words it is a control program used to manage user programs, prevent errors, protect resources, resolve conflicts.”¹ This type of technology is analyzed throughout this term paper. It includes three main sections, which are "Background Info", "Analysis" and "Conclusion".

“Background Info” section gives comprehensive feedback information about general technical issues, definitions of necessary methodology, and comprehensive explanations of operating systems used in today’s computer environment.

The second main section, which is the "Analysis" section, deeply analyzes the subject by using a variety of sub titles in order to efficiently cover the different aspects of

¹ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net>

the discussed issue. These sub titles includes "Necessity for an Operating System", "Technological Development and Evolution", "Existing Standards", and "Timeline". Overall this section of the paper focuses on the history, evolution, studies and practices of operating systems.

The last section is "Conclusion" section. This part of the paper tries to briefly summarize the general information given throughout the paper. Other than these three main sections, paper includes a comprehensive "Appendices" section, which provides many valuable figures, and tables. The content of these figures, and tables are closely related to the evolution of operating systems, practiced techniques, network structure designs, results of general surveys and technology information.

The main purpose of this paper is to provide efficient and accurate amount of information about the roots of current standards, the technical aspects, the actual use, technological improvements, and implemented modifications of this technology.

2- Background Info

First of all, some major vocabulary, methodology, important technical definitions and infrastructure issues of operating systems should be underlined in order to provide a general idea and to increase the knowledge about this technology. Operating systems are the results of a need for an important technology that enables people to work more efficiently and more productively with their computers. A serious definition of these systems can increase the understanding and be very productive in terms of exploring the issue. "One of the early studies of programming defines operating systems as the complex of programming, debugging, and operational aids with which the programmer

deals.”² Of course this kind of definitions were created during the 1960’s when there was not an established operating system industry or standards. Today there are more comprehensive, up to date, and explanatory definitions. An operating system can be defined as the basic software that controls and coordinates all the other parts and components of a computer. In his book called *Operating Systems, Internals and Design Principles*, William Stallings defines it as a program that controls the execution of application programs and acts as interface between the user of a computer and the computer hardware.”³ This also means that these systems provide a kind of communication between the end user and the computer. Before the creation of operating systems this kind of communication was very difficult if, especially in some cases, is not possible at all. In fact, they have six major objectives, which are “coordinating physical components of the computer, properly handling the organization of files by using variety of storage media, providing convenience for user, controlling the efficiency of the computer’s resources, ability to continuously improve the system and managing hardware errors and the loss of data.”⁴ All these objectives are important and have important roles for productive computer usage.

Evolution of operating systems brought these stated objectives because of certain necessities. These necessities are usually related to many different types of computer users. These computer users can be regular end-users, application programmers, or even operating system programmers (See Figure 1). As stated earlier, the reality is that they all need some kind of efficient communication solution with the hardware components of computers. This efficient communication way didn’t exist during the early years of

² Rosen, Saul, “*Programming Systems And Languages*”, New York: McGraw-Hill Book Company, 1967. (Page 517)

³ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997 (Page 45)

⁴ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997 (Page 45)

computing. People were dealing directly with the hardware to manipulate the functions of computers for even simplest activity. Operating systems were designed to overcome this lack of communication channel. Although the intention was revolutionary, early versions of operating systems were still very hard to use and to communicate. But they kept getting better over time with the help of many different universities, individual scientific efforts and investments of big corporations.

Over the years, two types of operating systems have been developed. One of these operating systems is text oriented (character driven). This type of operating system is usually used by advanced users. As it is text based, commands have to be typed. That is why it is also called as CUI (Command User Interface). UNIX and MS-DOS that will be covered later in details are good illustrations of this type. The other type uses Graphical User Interface (GUI). This type of operating system provides a virtual workspace, which is very easy to control, and allows user to communicate by clicking and pointing at virtual icons. “The first graphical user interface was developed by Xerox and used a new input device called a mouse.”⁵ Many of the most popular operating systems still use the technology that was developed by Xerox. These include GUI of Windows and Apple Macintosh. There are no commands to memorize or long procedures to follow. That is why it is highly common and popular among especially regular end-users. Today, the operating systems commonly installed on personal computers (PCs) are different models and versions of UNIX, Macintosh OS, MS-DOS, and Windows. On the other hand, there are other types of operating systems, which are mostly used for commercial use. These are usually designed and customized for specific professional machines. IBM’s AS/400 is one of these machines. “This machine has been around for 13 years, has its own

⁵ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

operating system and database built in, and is used by 250,000 companies, most of them manufacturers, to help manage their bread-and-butter operations.”⁶ Although the operating systems of these types of machines are not compatible with other more popular types of operating systems like Windows or UNIX, they are reliable and still do very well in the market.

3 - Analysis:

A - Necessity for an Operating System:

Every technology has its own advantages and disadvantages. And all these disadvantages create problems, which then generate necessities for different parts of human life. Computers and their technology are not an exception to this common idea. From the beginning of the computer usage, people had hard time to communicate, control and manipulate computer hardware. It was a difficult and an unpleasant experience to even accomplish the simplest tasks that are taken for granted today. Concepts like flexibility or convenience were not even the subject of a discussion about computers. The major expectation of the user was finishing the required task. And this was a long journey of waiting, struggling and suffering in front of gigantic machines. This inefficiency was affecting the demand for computers. As a result, this problem created a clear necessity for a new entity in computer industry. Operating systems are designed because of this necessity created by the problems of computer’s system architecture. A quick look at the definition of system architecture of computers might be useful in order to better understand the reasons of this necessity.

⁶ Bartholomew, Doug, “And it keeps on computing”, *Industry Week*, Issue 13, (Oct 2001), Page 73.

System architecture is the technical structure of computers that enables them to communicate with their components and to produce desired outcome. A simple but more sufficient and informative definition of system architecture of computers is presented in the article called “Turning points in systems architecture”, which was written by A. G. Genek and E. H. Sussenguth. “A system architecture is virtually any collection of computers and related computing resources and enabling software, however geographically dispersed, that are interconnected in some coordinated way so as to perform some useful function.”⁷ At the beginning, the system architecture of computers had many problems in terms of functionality, convenience and efficiency. Experiencing these problems encouraged many authorities to focus on the different ways of controlling the computer hardware. The result of this struggle was operating systems. Operating systems were designed to improve the productivity level of computers’ system architecture and later became part of it.

B - Technological Development and Evolution:

Looking at the evolution and development of operating systems is very productive in terms of understanding the values of achievements. Today, each computer has an operating system that provides a smooth and efficient computer experience. Many tasks, which were considered very difficult and demanding during 1950’s, 1960’s and even 1970’s, are performed easily without any hassle. The reason behind this improvement is in fact not only software improvements but more importantly it is the hardware development. Today’s hardware technology allows computer manufacturers and software

⁷ A. G. Ganek, E. H. Sussenguth, “Turning points in systems architecture”, *IBM Systems Journal*, Volume 38, Issue 2/3 (1999), Page 335-339.

designers wide variety of options with less technical limitations. The fact is that software development techniques or knowledge has never been fundamentally changed since its creation while hardware development has gone through an enormous renaissance. The logic and mathematical equations of programming had always been obvious to the programmers but they didn't have the tools (hardware) to realize the desired output. As operating systems can be considered as software, they are also closely related to the hardware and their designs and developments are limited by the hardware constraints. That is why an analysis of the operating systems' history would be very informative and extremely productive in terms of not only understanding the software timeline but also hardware development.

Early computers of 1940's and early 50's were bare machines that didn't have any type of operating system. Their appearance and the way they were operated were very primitive comparing to the current standards. "These machines were run from a console, consisting of display lights, toggle switches, some form of input device, and a printer."⁸ Operators had to directly deal with the hardware components and to manipulate them in order to get the desired outcome. Most of the programming languages that are taken for granted today simply didn't exist during those days. Operating any type of computer required many sets of detailed activities that took long hours. In addition, this whole process was not reliable at all. Command lines that are supposed to be entered in order to manipulate the computers might be mistakenly entered wrong or any hardware component might be malfunctioning. "For instance the usual mode of operation was for the programmer to sign up for a block of time on the signed up sheet on the wall, then come down to the machine room, insert his or her plugboard into the computer, and

⁸ William Stallings, "Operating Systems", New Jersey: Prentice-Hall, Inc., 1997. (Page 50)

spend the next few hours hoping that non of the 20,000 or so vacuum tubes would burn out during the run.”⁹ That was hard work and not a productive or efficient way of computing. “The se bare machines had two main problems, which were scheduling problems and setup-time problems.”¹⁰

Scheduling was time-consuming and inefficient. In order to use the computer, end-user had to guess the specific time period that he/she needed at the beginning of the task. After guessing the needed time period, he/she had to submit the job. If task took longer than expected, operator had to stop. If the task was finished before the signed-up amount of hours, the computer was idle until the preset time period ends. This process required a lot of time, spent the valuable work hours of operators and increased the computer idle time. Comparing to the modern ways of computer usage, this problem is highly primitive but it was a major issue of its day.

On the other hand, setup problems were related to the setup process itself. Setup process was long and had to be strictly followed before realizing any kind of computer related activity that is called “job”. This process had many steps and took long time. “Required steps were usually related to mounting or dismounting tapes or setting up cards decks.”¹¹ All these steps were preparing the computer to configure itself and its resources according to the desired functionality. A sample set of these required steps is presented in the book called “Operating System Concept”, which was written by Avi Silberschatz, Peter Baer Galvin, and Greg Gagne. “They state these steps of a common job as follows:

- 1- Loading the compiler tape

⁹ Tanenbaum, Andrew S., “*Modern Operating Systems*”, New Jersey: Prentice-Hall, Inc., 2001. (Page 7)

¹⁰ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 50)

¹¹ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 50)

- 2- Running the compiler
- 3- Unloading the compiler tape
- 4- Loading the assembler tape
- 5- Running the assembler
- 6- Unloading the assembler tape
- 7- Loading the object program
- 8- Running the object program”¹²

If any problem occurred during this setup process, operator had to begin all over again and go through all phases from the beginning. That is why computers were not practical or efficient for many types of tasks. In addition, it was a big burden to use computers for business purposes because of this problem. An organization that was willing to make computer purchases in order to transform its business activities into computerized operations also needed high level of utilization to compensate its investment. In order to improve these problems, operating system concept was considered by many leading authorities. As a result, this concept was fully explored. The fruits of this exploration came in mid 1950's. “First operating system in the history of computing was developed in the mid-1950's by General Motors for use on an IBM 701.”¹³ IBM saw the opportunity and began to work on this to improve and to make the necessary development. After the first operating system, many different modifications were made on this first operating system by many different people. As a result of these modifications, different early operating systems and some control programs were developed (See Timeline). Some of these are “ADMIRAL, Atlas I Supervisor, B1, B2, B3, B4, Basic Executive System,

¹² Silberschatz, Avi, Galvin, Peter Baer, Gagne, Greg, “*Operating System Concepts*” John Wiley & Sons, Inc., 2001. (Page 790)

¹³ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 51)

FMS, GM OS, HES, IBM 1410/1710 OS, Input Output Selector, Input Output System, IOCS, SABRE, SAGE and SOS.”¹⁴

Before going any further, the importance of IBM’s role in the evolution of operating systems should be underlined. One might say that IBM’s support for new ideas and struggle for better standards created an operating system industry and allowed many individuals and other new companies to get into this market. Although this statement may seem like an exaggeration, there are many examples from real life to prove it. “Gary Kildall who designed the CP/M (Control Program for Microprocessors) in 1974, was a good example of IBM’s support for new ideas. This operating system first ran on the revolutionary Intel 8080 8-bit processors.”¹⁵ Although, at the beginning, Kildall mostly designed his projects by himself, later on, IBM gave him an opportunity to share his views and ideas in a more productive and professional environment.

As stated earlier, after the first attempts of creating an operating system during 1950’s, IBM had constantly invested into this new software technology. This constant investment and close attention of IBM for operating systems encouraged young entrepreneurs and created a stable scientific environment for the creation of many important improvements like multitasking (See Time-sharing). For instance, IBM was the key player behind the creation of Operating System/2 (OS/2). “Microsoft and IBM jointly created Operating System/2 (OS/2) 1.00 which was the first OS that intrinsically supported multitasking based on hardware support.”¹⁶ If IBM didn’t support projects like these, overall development of operating system technology would be much more longer.

¹⁴ Mooney, Jim, (2001). Early Operating Systems. [Online]. Available: <http://www.csee.wvu.edu/~jdm/classes/cs258/cs258.html>

¹⁵ Anonymous, (2001). Operating Systems’ Quick History. [Online]. Available: <http://tsiolkovsky.cis.hcc.cc.il.us:100/cbrown/Fall%201999/CTEC%20150-01/Lecture%20Notes/Lecture%201/Default.htm>

¹⁶ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

After the initial design of this new operating system, Microsoft changed focus and began to develop early prototypes of Windows, which was heavily based on OS/2 structure.

“This change in Microsoft’s focus occurred in 1990.”¹⁷ This is especially true in terms of graphical user design. Later on, IBM continued to work on the project and came up with OS/2 1.x and OS/2 2.00. In addition, IBM designed many different customized operating systems, which are mostly used for on or two specific machines. IBM’s iSeries including mentioned AS/400 all have their own operating systems.

In other words, Microsoft got its education from IBM. This education generated a valuable training experience and knowledge base for Microsoft in terms of project management, operating systems design and development. As a result, company was able to get into the market by itself. That is why; Microsoft is another good illustration of IBM’s struggle for better standards in operating system development. The details of Microsoft’s story clearly show that IBM played an important role in the success of this company. Even before Bill Gates and Paul Allen made Microsoft name popular, “in 1980, they were hired by IBM to work on operating systems.”¹⁸ Before 1980, Microsoft was mostly dealing with issues like modifications of Basic and COBOL (Common Business-Oriented Language). After the intense interaction with IBM, Microsoft had enough experience to go further and began to work on its own operating system projects. “In 1981, Bill Gates visited Seattle Computer Products and bought the rights of Quick and Dirty Operating System (Q-DOS). In addition, he hired the creator of Q-DOS, Tim Patterson.”¹⁹ Tim Patterson was important for Microsoft because he was the programmer

¹⁷ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

¹⁸ Anonymous, (2001). Computer Timeline. [Online]. Available: <http://scils.rutgers.edu/~net99/comptime.html>

¹⁹ Anonymous, (2001). Operating Systems’ Quick History. [Online]. Available: <http://tsiolkovsky.cis.hcc.cc.il.us:100/cbrown/Fall%201999/CTEC%20150-01/Lecture%20Notes/Lecture%201/Default.htm>

of Q-DOS. Any development, which was related to Q-DOS had to be done under the supervision of a knowledgeable person about the issue. Microsoft used Q-DOS as a beginning point for its future projects. Company began to make modifications on the operating system (Q-DOS). The result of these modifications was MS-DOS. This modified version of Q-DOS became the basis for Windows operating system. Following the first versions of Windows, “Company released Windows 3.0 in 1990.”²⁰ This was a big break for Windows. Windows 3.0 became very popular. On the other hand, IBM continued to OS/2 series. “In 1992, IBM released OS/2 2.00, which was the first 32-bit OS for PCs.”²¹ After IBM saw the development potential of OS, management decided to work on its new versions like OS/2 Wrap, Wrap connect, Warp Server, and Warp 4 while Microsoft was realizing other projects like Windows NT, Windows NT 4, Windows NT 5, Windows 2000, Windows 95, Windows 98, Windows 98 SE, Windows Millennium (ME) and Windows XP. In fact, most of these different operating systems have the same roots with their own kind. Especially Microsoft products have always been modified versions of older ones. For instance, Windows 2000 is based on Windows NT and Windows ME is designed with fundamental structural techniques of Windows 95. As a result, a new and very strong industry occurred. Today, this industry includes wide variety of companies and their products. Although there are still standard problems, this industry will eventually improve itself. Of course this statement is true, if the competition can be established for its market.

Although early operating systems were useful and revolutionary, hardware and language constraints limited the efficiency level of designs and valuable technical

²⁰ Anonymous, (2001). Operating Systems' Quick History. [Online]. Available: <http://tsiolkovsky.cis.hcc.cc.il.us:100/cbrown/Fall%201999/CTEC%20150-01/Lecture%20Notes/Lecture%201/Default.htm>

²¹ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

features. The technology needed for better hardware was either unavailable or very expensive. Computer components were as big as rooms and weighted tons.

“Programming languages that were available were languages like Fortran and COBOL.”²² Even programming a simple function was time consuming and might not give the desired output. All these problems of computer industry created many problems and decreased the quality level of products. Throughout the evolution of these systems, many different techniques and approaches were used. All these different approaches created different operating system types.

“Simple Batch Systems”, “Multiprogramming Batch Systems”, and “Time Sharing Systems” are all part of the operating systems evolution. Although there are many other different types of them with small differences, these titles successfully categorize the main development cycle.

First generation operating systems were “Simple Batch Systems”. Second generation operating systems were called “Multiprogramming Batch Systems”. Next generation operating systems were “Time Sharing Systems”. Following subsections of the paper focuses on these three different types of operating systems and analyzes their technical and development aspects.

Simple Batch Systems:

During 1950’s and 1960’s, operating systems didn’t have colorful graphical interfaces that ease the control of computer. These early version of operating systems were very primitive and used only for limited amount of vital tasks. They were called “Batch Systems”. As stated earlier first of its kind was developed by General Motors. It

²² SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

was used to operate an IBM 701. This operating system then was reviewed and redesigned by many different IBM computer consumers. In addition, many other companies and programmers developed different brands of batch systems. One of the most successful of these early batch systems was IBSYS. “It was the operating system for the 7090/7094 computers.”²³ “This operating system actually was the predecessor of IBM OS/360 for 709x architectures and was developed during the early years of 1960’s.”²⁴ It was widely used for computers like 704, 709, 740 and 790. In fact during these years, many different computers used to handle different types of tasks. Computers like 7094 was used for more real computing tasks where computer like IBM 1401 was used for reading cards, copying tapes and for printing out (See Figure-4). “A regular procedure for finishing a task by using multiple machines included certain steps like:

- 1) Programmers were bringing the cards to 1401.
- 2) 1401 reads batch of jobs onto tape.
- 3) Operator carries input tape to 7094.
- 4) 7094 does computing.
- 5) Operator carries output tape to 1401.
- 6) 1401 prints output.”²⁵

The structure of simple batch operating systems was simple yet very productive. This type of operating system used a component that is called “monitor” (See Figure-2). “The monitor is system software that is responsible for interpreting and carrying out the

²³ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 51)

²⁴ Mooney, Jim, (2001). Early Operating Systems. [Online]. Available: <http://www.csee.wvu.edu/~jdm/classes/cs258/cs258.html>

²⁵ Tanenbaum, Andrew S., “*Modern Operating Systems*”, New Jersey: Prentice-Hall, Inc., 2001. (Page 8)

instructions of the batch jobs.”²⁶ The process begins with the submission of jobs (program or many programs) to the monitor. This submission, which usually was in the form of punched cards into the input device of computer, was handled by an operator. This process of job submission solved the scheduling problems. The main idea behind this solution is the gained ability to get rid of interventions and lose of idle time. Jobs could be processed sequentially without any interruption.

During the later years of batch systems, there were other developments, which successfully modified this type of operating systems. These modifications can be defined under four titles. First of these modifications was the “memory protection”²⁷ that allowed monitor to identify and separate the outputs of different jobs. This way jobs didn’t affect each other. Secondly, a “timer”²⁸ is added into the system. This timer controls the durations of jobs and protects the process from infinite loops. This feature actually is the first development in operating systems that later provide a technical basis for the creation of multiprogramming systems. “Privileged instructions”²⁹ were another modification that was made in the structure of simple batch operating systems. These instructions were highly useful in terms of controlling and organizing the output of different jobs. “Final type of modification was the interrupts that gives the operating system more flexibility in relinquishing control and regaining control from user programs.”³⁰

²⁶ Gordon College. (2000). A Brief History of Computer Operating Systems. [On-Line]. Available: <http://www.math-cs.gordon.edu/courses/cs322/lectures/>

²⁷ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 53)

²⁸ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 53)

²⁹ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 53)

³⁰ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 53)

Multiprogramming Batch Systems:

During the mid 1960's, operating systems began to evolve and take a different shape. The main reason of new developments was the decreasing cost of memories. This evolution produced a new generation of operating system, which is called "Multiprogramming Batch System". "As computers with more memory became increasingly available, it became possible to load a number of jobs into memory (from the job pool on disk) at the same time and the operating system now became responsible for job scheduling."³¹ The structure of multiprogramming batch operating systems was based on the prevention of idle time and the unnecessary utilization of processor. This new type of operating system was developed to efficiently handle multiple tasks altogether.

"In other words multiprogramming creates systems that would load several jobs into memory at once and cycle through them in some order, working on each one for a specified period of time."³² For instance, if one task needs time to be executed due to the speed limitations of input/output devices, this task is replaced by another task. This provided enormous time efficiency and allowed processors to work on multiple jobs at the same time without interrupting any specific process (See Figure-3).

Multiprogramming is a highly important development because its structure created a foundation for today's advanced operating systems. One of the earliest multiprogramming operating systems was OS 360. "IBM 360 and several systems were

³¹ Professor Smith, (August 28, 2000). [Online]. Available: http://www-inst.eecs.berkeley.edu/~cs162/Students/Lecture_Notes_00.f/08-28-00_1.txt

³² Gordon College. (2000). A Brief History of Computer Operating Systems. [On-Line]. Available: <http://www.math-cs.gordon.edu/courses/cs322/lectures/>

equipped with a special hardware to protect each job against snooping and mischief by the other jobs.”³³

Time Sharing Systems (Multitasking):

In 1961, first time-sharing operating system was developed. “Time sharing means the ability of a computer to run more than one program at a time.”³⁴ The standards of mainframe computer usage completely changed with this multiple program execution capability. That is why, this development was on time and very important in the evolution of operating systems.

“Time-sharing system development was made at MIT by a group known as Project MAC (Machine Aided Cognition, or Multiple-Access Computers).”³⁵ This time-sharing operating system was called CTSS and was prepared for IBM computers (first for IBM 709 and second for IBM 7094). A general comparison between multiprocessing batch systems and time-sharing systems can be seen in Table-2, which is in the “Appendices” section of this paper.

Time-sharing operating systems increased the productivity level of mainframe computers by using features like priority and interrupts. “Priority has to deal with which processes take precedence over others (Interactive processes and those that don't take much CPU time are usually given higher priority to ensure they get done.), and interrupts were basically signals to the CPU to tell it that an I/O event has occurred.”³⁶ In fact, they allow multiple users to interact with mainframe computer at the same time. In other

³³ Yu, Byunggu, (2001). [Online]. Available: <http://www.cs.uwo.edu/~yu/COSC4740-01/Lecture1.pdf>

³⁴ Walters, E. Garrison, “*The Essential Guide to Computing: The Story of Information Technology*”, New Jersey: Prentice-Hall, Inc., 2001. (Page 162)

³⁵ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 58)

³⁶ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

words, several users could share the resources of the computer simultaneously by sharing the CPU time. As a result, “even though there was only one central computer, to the users it seems that they each had their own independent computer.”³⁷ The stated feature of time-sharing operating systems is its major difference than earlier types of operating systems. This was highly important and revolutionary because mainframe computer with only one processor were the only option for computer-related tasks of many different end-users. There was not client server architecture where many personal computers with their own processors could be interconnected and individually used.

Today, this technique is still widely used in the structure of operating systems and there are many different types of them. “One important type is represented by computers like VAX/VMS or UNIX workstations in which entire processes are in memory (albeit virtual memory) and the computer switches between executing code in each of them; in other types of systems, such as airline reservation systems, a single application may actually do much of the timesharing between terminals.”³⁸

VAX/VMS is a commercial system like IBM’s AS/400. It is a product of Digital Equipment Corp (DEC). “Its initial development was finished in 1975 and first shipped in 1977.”³⁹ This system has been very efficient and cost-effective for many companies.

“According to the study, the AS/400 and VAX systems, because of their strong systems management capabilities, cost just \$36 per user per month in this category, roughly one-tenth the cost of NetWare.”⁴⁰ In addition, DEC developed another similar product, which

³⁷ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

³⁸ Gordon College. (2000). A Brief History of Computer Operating Systems. [On-Line]. Available: <http://www.math-cs.gordon.edu/courses/cs322/lectures/>

³⁹ Anonymous, (2001). Operating Systems’ Quick History. [Online]. Available: <http://tsiolkovsky.cis.hcc.cc.il.us:100/cbrown/Fall%201999/CTEC%20150-01/Lecture%20Notes/Lecture%201/Default.htm>

⁴⁰ Cummings, Joanne, “Study finds AS/400 to be cost-effective net platform”, *Network World*, Volume 10, Issue 11, (Mar 15, 1993), Page 25.

was called Ultrix. This operating system is based on UNIX and has been used for many commercial systems. “The company supported its own Ultrix (DEC’s Unix operating system) on the R4000-based DECstations and DECsystems.”⁴¹

On the other hand, “UNIX, which was first developed in 1969 by a research group at Bell Labs and was based on the MULTICS project”⁴², became very popular among users later on. This popularity made this operating system to be widely used. In fact, the main purpose of this project was to develop a tool that will be used for academic studies. “After four year of silence, in 1973, UNIX was modified and rewritten in C.”⁴³ “In fact, design of C was completed by Dennis Ritchie one year earlier in 1972 at Bell Labs. This language will become the standard in microcomputer.”⁴⁴ This modification made the system more stable and reliable. Its reliability and functionality attracted the attention of many other authorities from different environments. Today there are many types of operating systems that are based on UNIX. “Sun Microsystems Solaris, UNIX System V, SGI's IRIX, and BSD can be good illustrations of these.”⁴⁵

Of course Linux also has to be mentioned when the subject is related to UNIX because they have many similarities. “Linux was first developed for the Intel 80386 in 1991 by Linus Torvalds and it was especially designed to be compatible with UNIX.”⁴⁶ Although Linux was first developed to be an independent operating system, its structure and technical aspects are closely related to UNIX. Over the years, Linux became a succcessfully-modified version of UNIX. As Linus Torvalds preferred to make Linux public

⁴¹ Johnson, Maryfan, “DEC Flip-Flops on Unix Workstation Strategy”, *Computerworld*, Volume 26, Issue 24, (Jun 15, 1992), Page 1.

⁴² SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

⁴³ Sigler, Lee, “Unix’s Infancy: An Operating System Grows at Bell Labs”, *Computerworld*, Volume 18, Issue 39, Sep 26, 1984, Page 75.

⁴⁴ Anonymous, (2001). Computer Timeline. [Online]. Available: <http://scils.rutgers.edu/~net99/comptime.html>

⁴⁵ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

⁴⁶ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

through Internet, its development has been very stable. As a matter of fact, this way of development that is open to everyone increased the speed and the quality of Linux's evolution. Today, Linux is considered by many people as an extension of UNIX. There are many different versions of Linux too. Linux-Mandrake by Macmillan, Red Hat Linux by Red Hat, Caldera OpenLinux and SuSE Linux by SuSE, Inc. are some of these versions. Each one of these different Linux clones has its own advantages and disadvantages. "For instance, Caldera is ideal for secure remote administration, but it lacks some easy setup tools for firewall and local graphical managing tools. Mandrake is probably the easiest tool for those making a transition from Windows. It provides a lot of similar administrative tools and a GUI that makes Windows users feel right at home. Red Hat is a good overall package with good documentation, and its RPM format certainly makes installing new software easier because it's still one of the most widely used Linux distributions, there's better feedback on bugs, as well as quicker patches. SuSE's Linux Professional offers the best security software and comes with not only two software firewalls, but also a well-written system configuration manual. SuSE also has a managing tool that's similar to the Windows Control Panel."⁴⁷

C - Existing Standards:

Today, there is an enormous growth in the operating system market that is driven by the high demand for better products. "For instance, the Linux market stands to double this year to \$4 billion, according to Deutsche Banc Alex. Brown, a Wall Street

⁴⁷ Kuo, Joshua, "Operating Systems- Work Ready Linux—Easy management, solid security top benefits of four Linux distributions", *Internetweek*, Issue 877, (Sep 10, 2001), Page 23.

investment firm.”⁴⁸ This kind of growth and related revenues will exponentially increase the speed of the operating system evolution. Operating system industry has been struggling to catch up with the customer demand. In order to make drastic changes, to shape their industries and to increase their sales, most of the developers began to focus on new techniques (See Table-1). “Some of these fresh techniques are microkernel architecture, multithreading, symmetric multiprocessing, distributed operating systems, and object-oriented design.”⁴⁹

All of these techniques are still not fully developed and are in testing phase. In addition, developers of these new types of operating systems have not been able to come up with drastic changes. Most of these new systems have no or little differences. “For instance, a multiprocessor system has shared memory and hardware between processors while in a distributed system, each processor has it's own memory environment.”⁵⁰ Operating system developers of today’s computer environment including Windows, Unix, and etc. have an opportunity to make changes. This opportunity is based on the availability of state of the art hardware technology. In addition, this hardware technology is not expensive anymore. This means that developers will not have the limitations or constraints of hardware when they generate new approaches to the existing problems of operating systems.

⁴⁸ Rooney, Paula, “Operating without liability”, *Computer Reseller News*, Issue 930, (Jan 29, 2001), Page 14.

⁴⁹ William Stallings, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997. (Page 72)

⁵⁰ SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>

D - Timeline:

The following timeline shows the release dates of different operating systems. This timeline was created by using many different resources. Among these resources there are different websites, and articles. Although it is modified with extra information, major resource for this table is obtained from Osddata.com, <http://scils.rutgers.edu/~net99/comptime.html>, <http://tsiolkovsky.cis.hcc.cc.il.us:100/cbrown/Fall%201999/CTEC%20150-01/Lecture%20Notes/Lecture%201/Default.htm>, <http://www.csee.wvu.edu/~jdm/classes/cs258/cs258.html> and Sigler, Lee, “Unix’s Infancy: An Operating System Grows at Bell Labs”, *Computerworld*, Volume 18, Issue 39, Sep 26, 1984, Page 75.

Operating Systems’ Timeline

Mid-1950’s

- GM OS (Unnamed) - First operating system was developed by General Motors for an IBM 701 (Probably 1955). This development attracted the IBM’s attention. Company began to invest into this new concept.

Late-1950’s

- FMS - FORTRAN Monitor system. Operating System developed by North American Aviation for the IBM 709.
- Input Output System - A very early operating system developed by General Motors and North American Aviation for the IBM-704. (Around 1956).
- SAGE - Semi-Automatic Ground Environment system. Control Program for IBM AN/FSQ7 to monitor weapons systems. First real-time control system.
- SOS - Operating System was developed by the IBM SHARE Users Group for the 709.

Early-1960’s

- ADMIRAL - Operating system for the Honeywell 800/1800.
- B1, B2, B3, B4 - Operating Systems for NCR Century series computers.
- Basic Executive System - Memory-based real-time executive for the IBM 1710.
- BOS/360 - Basic Operating System for the IBM/360.
- CTSS - First time-sharing operating system was developed at MIT for a modified IBM 7090. (1961).
- EXEC I, EXEC II - Single User OSs for UNIVAC 1107 computers.
- HES - Honeywell Executive System. Operating System for the Honeywell 800.
- IBM 1410/1710 OS - OS for the IBM 1410/1710.
- IBSYS - Predecessor to IBM OS/360 for 709x architectures.
- IOCS - Input Output Control System. An early I/O Executive for the IBM 7090. (About 1960).
- JOSS - JOHNNIAC Open Shop System. An early on-line computing system for the JOHNNIAC computer. Early 1960s.

- MCP - Master Control Program. Generic name used for Burroughs Operating Systems, including 1700 and 5500, 6600, and 7700 series.
- SABRE - Semi-Automatic Business Related Environment. The first major transaction processing system, developed by IBM and American Airlines for the IBM 7090.
- TSM - Time Sharing Monitor. Early timesharing OS developed by IBM for the 7090. Early 1960s.
- TSS - An early timesharing system for the AN/FSQ-32. Early 1960s.

Mid-1960's

- Atlas I Supervisor - Supervisor Program for the first computer designed to use an operating system. Introduced system calls and virtual storage. Location was University of Manchester.
- DOS/360 - Operating System for IBM S/360. IBM Corp.
- DTSS - Dartmouth Time-Sharing System. Timesharing with the BASIC language for liberal arts students, on a novel 2-computer system (GE-235 + Datanet 30). Mid 1960s.
- EGDON - Serial batch OS for the English Electric KDF9, designed to support program development. Mid 1960s.
- Input Output Selector - An IO Control System for the DDP-116 minicomputer. One of the earliest OSs for minis.
- PDP-6 Monitor - Monitor System for the DEC PDP-6. Mid 1960s.
- TITAN - Operating System for the Titan (Atlas II) Computer, successor to Atlas I at the U. of Manchester. mid 1960s.
- TOS/360 - Early operating system for the IBM S/360. Mid 1960s.
- TSOS - Time Sharing OS for RCA Spectra 70/46. Mid 1960s.
- TSS/360 - Timesharing OS developed by IBM for the S/360. Mid 1960s.

Late-1960's

- AMS - Advanced Monitor System. A single user interactive disk or tape OS for the PDP-9 & PDP-15. Digital Equipment Corp., Late 1960s.
- BPS/360 - Business-oriented OS for the IBM/360.
- BTM - Batch Time-Sharing Monitor for XDS Sigma 5/7. Late 1960s.
- CAL - Research OS for the CDC 6400.
- CMAS - Cambridge Multiple Access System. University of Cambridge (England). Late 1960s.
- CPS - An early OS for DEC minicomputers. Late 1960s.
- DOS - Disk Operating System for RCA Spectra 70 series.
- EGDON 3 - OS for the British KDF9. Based on EGDON. Late 1960s.
- ELDON 2 - Operating System for the British KDF9. Late 1960s.
- EXEC 3, EXEC 4 - Improved versions of EXEC II for UNIVAC 1108.
- EXEC 8 - Multiprogrammed OS for UNIVAC 1108.
- Foreground/Background/8 - Experimental Foreground/Background OS for the PDP-8. Late 1960s.
- GECOS III - Multiprogrammed OS for GE 600 series computers.
- George 1,2,3, and 4 - OS for various models of ICL 1900 series computers.
- IDA, IDASYS - OS for the CDC 6600 developed at the Institute for Defense Analyses, based on the concept of a supervisory computer.
- KOS - Kent On-Line System, Kent Univ. England, Late 1960s.
- MASTER - OS for Control Data 3300/3500.
- MINIMOP - Multi-Access OS. Late 1960s.
- MOD4 - OS for the Honeywell Series 200. Late 1960s.
- MOS - Multiprogramming OS for NCR Century 300 computer. Late 1960s.
- MTS - Michigan Terminal System. Late 1960s.
- MULTICS - The MULTiplexed Information and Computing Service. A timesharing system featuring novel techniques for memory management and resource protection. Developed by M.I.T., Bell Labs, & G.E. for the GE645. Later developed by Honeywell. Late 1960s.
- OS/200 - OS for Honeywell Series 200 family. Late 1960s.
- OS-3 - Oregon State Open Shop OS. A timesharing system for the CDC 3300. Late 1960s.

- OS/360 - Generic name for operating systems for the IBM S/360, and later S/370. First version released 1966. (OS/PCP, OS/MFT, OS/MVT, OS/VS1, OS/SVS, and OS/MVS are related to this operating system).
- OS/MFT, OS/MFT-II - Multiprogramming with a fixed number of tasks. Simple version of OS/MVT suitable for processors with limited memory. First release 1967, substantial revision 1968.
- OS/MVT - Multiprogramming with a Variable Number of Tasks. Popular OS for IBM 360/370. First release 1967.
- OS/PCP - Primary Control Program. Early batch-serial version of IBM OS/360. First release 1966.
- PDP-10/50 Swapping Monitor - An early OS for the DEC PDP-10. Late 1960s.
- Pitt Time Sharing System - An early Timesharing system developed at the University of Pittsburgh for the IBM S/360. Late 1960s.
- POS - Primary OS for RCA Spectra 70. Late 1960s.
- SCOPE - Supervisory Control of Program Execution. Early OS for CDC 6000 Computers. Late 1960s.
- SEL 810A/810B OS - Batch Serial Operating System for the SEL 810A/810B minicomputer. Late 1960s.
- SUE - Experimental OS for the IBM S/360. Late 1960s.
- TDOS - Tape and Disk operating system for the RCA Spectra 70. Late 1960s.
- TOPS-10 - Total Operating System. Timesharing OS for DEC PDP-10. Digital Equipment Corp. Late 1960s.
- TSS/8 - An early timesharing system for the AN/FSQ-32. Early 1960s.
- UNIX - A research group at Bell Labs developed UNIX. The project was called MULTICS. (1969).
- VMOS - Timesharing OS for the RCA Spectra 70. Late 1960s.

Early-1970's

- 2000E & 2000F - Timesharing operating systems for the HP 2100. Early 1970s.
- BCM - Basic Control Monitor for Xerox 530. Early 1970s.
- BCS - Basic Control System for HP 2100. Early 1970s.
- BEST - Basic Executive & Timekeeper. OS for Varian minicomputers. Early 1970s.
- BKY - A batch-oriented OS for the CDC 6600 at Lawrence Berkeley Laboratories.
- BMS - Basic Monitor System for the DEC PDP-9 and PDP-15. Early 1970s.
- BOS - Basic OS for Interdata (later Perkin-Elmer) Minicomputers. Early 1970s.
- BOSS - Basic OS Software for Basic/Four 350/400/500. Includes monitor, real-time executive, BASIC. Early 1970s.
- CAPS-8 - Cassette Monitor system for the DEC PDP-8. Early 1970s.
- CAPS-11 - Cassette Monitor System for the DEC PDP-11. Early 1970s.
- Chios - OS for the UNIVAC 1108 in a service bureau environment. Case Western Reserve.
- CiMOS - Batch oriented disk operating system for the Cincinnati Milacron CIP/2200. Early 1970s.
- COS - (THREE DIFFERENT OSs) Concurrent Operating System. Provided limited multiprogramming for Univac 9200, 9300 AND Operating System for GEC Computers. AND Cassette OS for Computer Automation Alpha LSI and Naked Mini LSI. Early 1970s.
- CP/M (Control Program for Microprocessors) - It was created by Gary Kildall. This operating system first ran on the Intel 8080 8-bit processors. (1974).
- CP/M's version for 8086 was also developed around the same time. (1974).
- CPPS - Card/Paper Tape Programming System for the IBM 1130. Early 1970s.
- DMS - Disk Monitor System for IBM 1130. Early 1970s.
- DOS-15 - Disk OS for DEC PDP-15. Early 1970s.
- DOS/VS - Operating System for the IBM S/370. Virtual Storage version of DOS/360. IBM Corp.
- Pipes and Filters for the C Language (In 1973 it will be used to rewrite UNIX) - Dennis Ritchie, who is also one of the creators of Unix, developed in Bell Labs, introduced pipes and filters for the C language. (1972).
- EDOS - Operating System for the Hitachi 8400. Early 1970s.
- FSOS - Free Standing Operating System. Basic memory-resident OS for General Automation SPC-16 mini. Early 1970s.

- GCOS - General Comprehensive OS. Honeywell Series 60 and 6000 systems. Based on GECOS. Early 1970's.
- KRONOS - Timesharing OS for CDC 6000 and CYBER computers. Successor to SCOPE. Control Data Corp., Later it will be used to create NOS. Early 1970s.
- MAX - Principal OS for Mod Comp Computers. Early 1970s.
- MCTS - Multiple Console Time Sharing System developed to support CAD applications at General Motors. Early 1970s.
- Min I - Basic Operating System for the ModComp Computers. Early 1970s.
- MOS - Minimum Operating System. Card-based batch monitor for Univac 9200, 9300. Early 1970s.
- MPE-3000 - OS for HP 3000. Early 1970s.
- MTDOS - Magnetic Tape OS for Datapoint 1100, 2200, 5500. Early 1970s.
- MTDS - Magnetic Tape OS for Computer Automation Alpha LSI & Naked Mini LSI. Early 1970s.
- MTS - Magnetic Tape System for HP 2100. Early 1970s.
- MU5 - OS for the MU5 computer at the U. of Manchester (?). Early 1970s.
- MUMPS - Multi-user multiprogramming, database oriented OS for DEC computers. Early 1970s.
- NCOS – Non-concurrent OS. Tape or disk based OS for Univac 9200 & 9300. Early 1970s.
- NUCLEUS - Operating System for GEC Computers. Early 1970s.
- OS6 - Experimental OS for a small computer. Early 1970s.
- OS/MVS - Multiple Virtual Space. Virtual storage OS for IBM 370, derived from MVT. Originally called VS2, Release 2. First release 1974.
- OS/SVS - Single Virtual Space. Initial release of OS/VS2. Virtual Storage OS for IBM S/370, based closely on OS/MVT. First release 1972.
- OS/VS1 - Virtual System 1. Virtual Storage OS for the IBM 370, based on OS/MFT. First release 1972.
- PTSS - People's Time Sharing System. A time-sharing system for the CDC 6600 at Lawrence Berkeley Laboratories, based on BKY. Early 1970s.
- Q-DOS - Once the 8086 was released a new operating system emerged, called QDOS. (1974).
- SCP - System Control Program. Basic OS for the Burroughs B700 series. Early 1970s.
- SOS - Standalone OS for Data General Nova & Supernova. Early 1970s.
- SYSEX - System Executive. A time-sharing monitor for SIGMA/2 and small IBM S/360 computers. Early 1970s.
- Tape Scope 2 - Tape based OS derived from SCOPE for the CDC System 17. Early 1970s.
- Tenex - A Timesharing System for a modified PDP-10. Based partially on abstract machine concepts. Early 1970's.
- TOPS-20 - Improved version of TOPS-10 for the DECSystem 20, also using concepts from TENEX. Early 1970s.
- TOS - Teletype Operating System for the Microdata 1600 & 3200 minicomputers. Early 1970s.
- TSS - An early timesharing system for the AN/FSQ-32. Early 1960s.
- UNIX was rewritten in C. This change made this particular operating system more reliable and stable. (1973).
- UTS - Time Sharing Operating System for Xerox Data Systems SIGMA computers. Early 1970s.
- VENUS - Experimental OS for a modified Interdata minicomputer, based on matching the computer architecture to the operating system. Early 1970s.
- VS/9 - Operating System for the Univac Series 90. Derived from VMOS. Early 1970s.

Mid-1970's

- BPS-99 - Business oriented operating system for the GRI-99. Mid 1970s.
- OS7 - Operating System for HITAC 8700/8800. About 1975.
- OS8 - Operating System for the PDP-8. Ancestor of RT-11. Digital Equipment Corp. Mid 1970s.
- RT-11 - Single User OS for DEC PDP-11. Digital Equipment Corp. About 1975.
- RDOS - Single-User OS for the Data General NOVA. Mid 1970s.
- SSP - System Support Program. Multitasking OS for the IBM System/34. Mid 1970s.
- VAX/VMS - Conception (June 1975).

Late-1970's

- AMIGO/300 - Small, user-friendly OS for the Hewlett-Packard 300. Late 1970s.
- AOS - Advanced Operating System for Data General Eclipse and MV series. Late 1970s.
- BOSS-15 - Batch OS for the DEC PDP-15. Late 1970s.
- COS300 - Commercial data management system for the PDP-8. Late 1970s.
- CTOS - Cassette Tape OS for Datapoint 1100, 2200, 5500. Late 1970s.
- DBOS - Disk/Drum Based OS for General Automation SPC-16 & System 18/30. Late 1970s.
- DOS-99 - Single User operating system for the GRI-99 Minicomputer. GRI Corp. Late 1970s.
- DOS - OS for Datapoint minicomputers. Late 1970s.
- EMAS, EMAS 2900 - Edinburgh Multi-Access System for ICL 4-75.
- MODUS4 - OS for Computer Technology minicomputers. Late 1970s.
- TRIPOS - A portable OS for Minicomputers. PDP-11, 68000 series, 32000 series. Late 1970s.
- TS8/E - Timesharing OS for the DEC PDP-8. Late 1970s.
- VAX/VMS - Standard operating system for DEC VAX. Based on RSX-11. First VAX Ship date (October 1977).
- VMS V1.0 - (August 1978).

Early-1980's

- AmigaOS 1.0 - It was announced. (1983).
- BOS/5 - Business Operating Software. Business oriented OS for Z80, PDP-11, 8086, and 68000. BOS National, Inc. Early 1980s.
- CPF - Control Program Facility. Operating System for the IBM System 38. Early 1980s.
- DBLDOS - OS for the Tandy TRS-80. Simple updates to TRSDOS to support the Percom double-density board. Percom Data Co. Early 1980s.
- DOS/VSE - Operating System for the IBM 43XX. Extended version of DOS/VS. Early 1980s.
- IBM focused on CP/M and Basic. Bill Gates and Paul Allen were hired by the company to work on a new operating system. (1980).
- Macintosh - (January 1984).
- MDOS - Operating System for the Data General MV4000. Early 1980s.
- P/OS - Professional Operating System. Standard OS for the PRO-350 and PRO-380. Derived from RSX-11. Digital Equipment Corp. Early 1980s.
- PrimOS - OS for the PRIME minicomputers. Early 1980s.
- Q-DOS (important development)- Bill Gates wanted to get into the operating system market. In order to realize this, he had to have his own OS. So, he visited Seattle Computer Products and bought the rights of Quick and Dirty Operating System (Q-DOS). In addition, he hired the creator of Q-DOS, Tim Patterson. This was particularly important in terms of developing and modifying Q-DOS. Microsoft developed it further, calling it MS/DOS. This operating system became the basis for future Windows versions. (1981).
- SUN - General Purpose OS kernel for the HP 9000 Series 500. Early 1980s.
- ULTRIX V1.0 - (1984).
- VAX/VMS 4.0 - MicroVMS announced with VAX/VMS 4.0. (December 1984).
- VMS V3.0 - (April 1982).
- VMS VAXCLUSTERS - It was announced. (1983).
- VMS V2.0 (April 1980).
- VMS V4.0 - (September 1984).

Mid-1980's

- AmigaOS 1.0 - (October 1985).
- VMS V4.2 - (1985).
- Windows 1 - (not successful) (1985).

Late-1980's

- HP-UX 1.0 - (1986).
- HP-UX 1.1 - (1987).

- HP-UX 1.2 - (1987).
- HP-UX 2.0, 2.1, 3.0 - (1988).
- HP-UX 3.1, 7.0 - (1989).
- OS/2 1.0 - (December). (1987).
- VAX/VMS 4.4 - MicroVMS retired with this new operating system. (December 1986).
- VMS V4.4 - (1986).
- VMS V4.7 - (1986).
- VMS V5.0 (May 1988).
- VMS V5.2 - (September 1989).
- Windows 2 - (not successful). (1987).

Early-1990's

- AIX 3.0 - (1990).
- BSDi BSD/OS - initial production release. (March 1993).
- FreeBSD 1.0 - (December 1993).
- FreeBSD 1.1 - (May 1994).
- HP-UX 7.02, 7.06, 7.08 - (1990).
- HP-UX 8.0, 8.01, 8.02, 8.05, 8.06, 8.06+, 8.07 - (1991).
- HP-UX 9.0, 9.01 - (1992).
- HP-UX 9.02, 9.03 - (1993).
- HP-UX 9.04, 9.05 - (1994).
- Linux - It was developed by Linus Torvalds. He made it public through Internet. (1991).
- OpenVMS ALPHA V1.0 - It was based on VAX/VMS 5.4 (November 1992).
- OpenVMS ALPHA V1.5 - (1993).
- OpenVMS ALPHA 6.1 - (May 1994).
- OpenVMS VAX V6.0 - (June 1993).
- OpenVMS VAX 6.1 - (April 1994).
- OS/2.00 - It was released by IBM. It was the first 32-bit OS for PCs. (1992).
- VMS V5.4 - (October 1990).
- VMS V5.5, 5.5-1, 5.5-2 - Later this year its name changed to OpenVMS (November 1991).
- Windows 3.0 - It becomes popular. Graphical User Interface is easy to use and practical. It sells more than 3 million copies during this year. (1990).
- Windows 3.1 - It is introduced. 1million copies of it were sold on one month. (1992).

Mid-1990's

- BeOS - "Be publicly shows the BeOS for the first time. At that time, Be builds a proprietary hardware called the BeBox (which is a dual PowerPC machine, roughly a boosted PReP machine)." (October 1995).
- FreeBSD 2.0 - (January 1995).
- FreeBSD 2.0.5 - (June 1995).
- HP-UX 9.07, 10.0, 10.01 - (1995).
- OpenVMS 7.0 - (December 1995).
- OpenVMS ALPHA 6.2 - (June 1995).
- OpenVMS VAX 6.2 - (May 1995).
- ULTRIX V4.5 - (November 1995).
- Windows 95 - is introduced to the market. (1995).

Late-1990's

- AIX 4.3 - (October 1997).
- AIX 4.3.2 - (October 5 1998).
- AIX 4.3.3 - (October 1999).
- BeOS - Advanced Access Preview Release (May 1997).
- BeOS Dr6 - (developer release) (January 1996).

- BeOS Dr7 - (developer release) (April 1996).
- BeOS Dr8 - (developer release) (September 1996).
- BeOS PR - (preview release) (July 1997).
- BeOS PR2 - (preview release) (October 1997).
- BeOS Release 3 - for Intel x86. (March 1998).
- BeOS Release 3 - for PowerPC. (April 1998).
- BeOS Release 3.1 - (June 1998).
- BeOS Release 3.2 - (July 1998).
- FreeBSD 2.1.5 - (August 1996).
- FreeBSD 2.1.7.1 - “end of mainstream development on 2.1-stable.” (February 1997).
- FreeBSD 2.2 - “branched from the development mainline.” (November 1996).
- FreeBSD 2.2.1 - “first full release of 2.2 [series].” (April 1997).
- FreeBSD 2.2.7 - (July 1998).
- FreeBSD 2.2.8 - “the last release on the 2.2 branch.” (November 1998).
- FreeBSD 3.0 - “first official 3.0 release.” (October 1998).
- HP-UX 10.10, 10.20 - (1996).
- HP-UX 10.30, 11.00 - (1997).
- OpenVMS 7.1 - (December 1996).
- OS/2 4.0 - (September 1996).
- Mac OS 8 - is introduced to the market. (1997).
- Macintosh 8. 5 (October 1998).
- Windows 98 - is introduced to the market. (June 25 1998).

Early-2000's

- HP-UX 11.11 - (aka 11i). (2000).
- Macintosh OS X public beta - (September 13, 2000).
- Macintosh OS X 10.0 - (March 24, 2001).
- Windows 2000 1.0, Windows Me - (February 17, 2000).
- Windows XP - is released. Some new features like Passport system is added. Microsoft claims that is more reliable and stable than older Windows versions. (2001).

4 - Conclusion:

In conclusion, operating systems are highly important for today's computer environment. They became one of the most important components of modern computers and they are vital. The creation and evolution of operating systems began with the necessity of user friendliness. The lack of communication channels between the user and computer hardware pushed many organizations and institutions to focus on this issue. As a result operating systems facilitated many tasks and activities that are related to computers. In addition, this evolution established a strong industry that can infinitely focus on the important issues in order to compete in business arena. This competition will soon bring many more developments and major changes that will increase the productivity level of business and social environment.

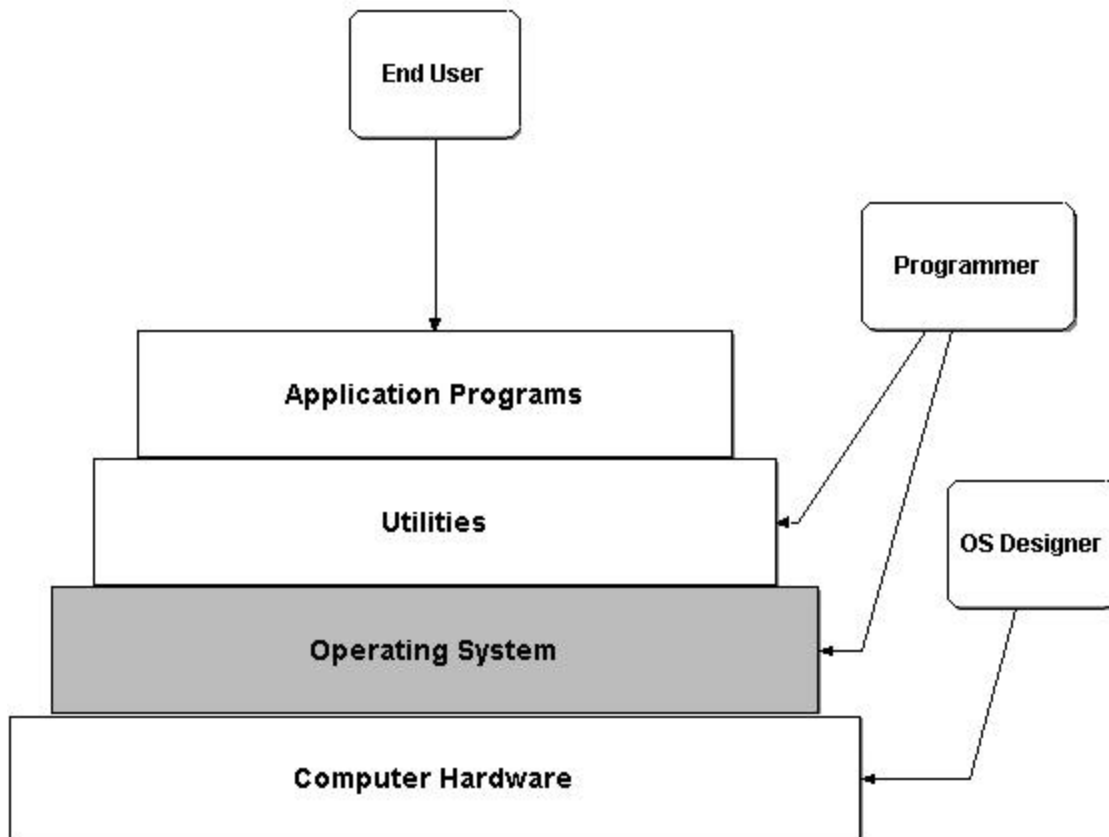
Future generation operating systems will be user-friendlier and allow increased level of interactivity. Existing developments are just the beginning of a magnificent virtual era.

5 - Appendices:

Figures:

Figure-1⁵¹

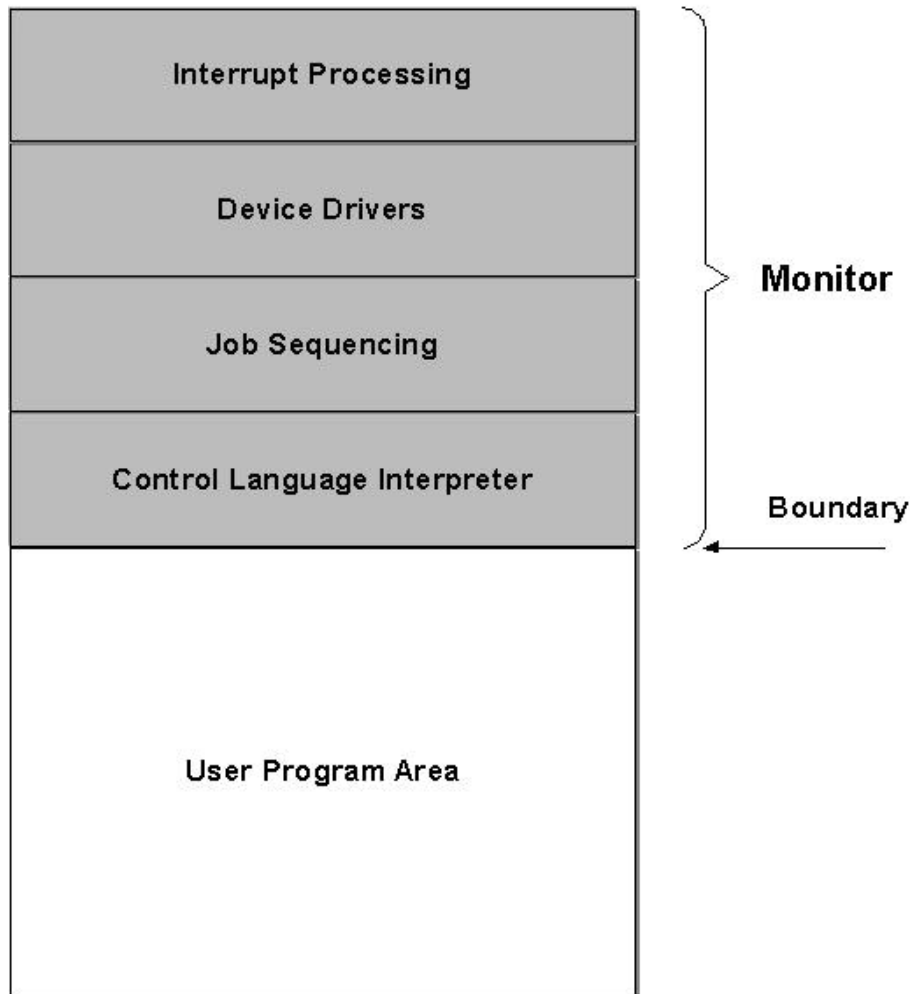
Different Parts of Computer Structure



⁵¹ William Stallings, "Operating Systems", New Jersey: Prentice-Hall, Inc., 1997. (Page 46)

Figure-2⁵²

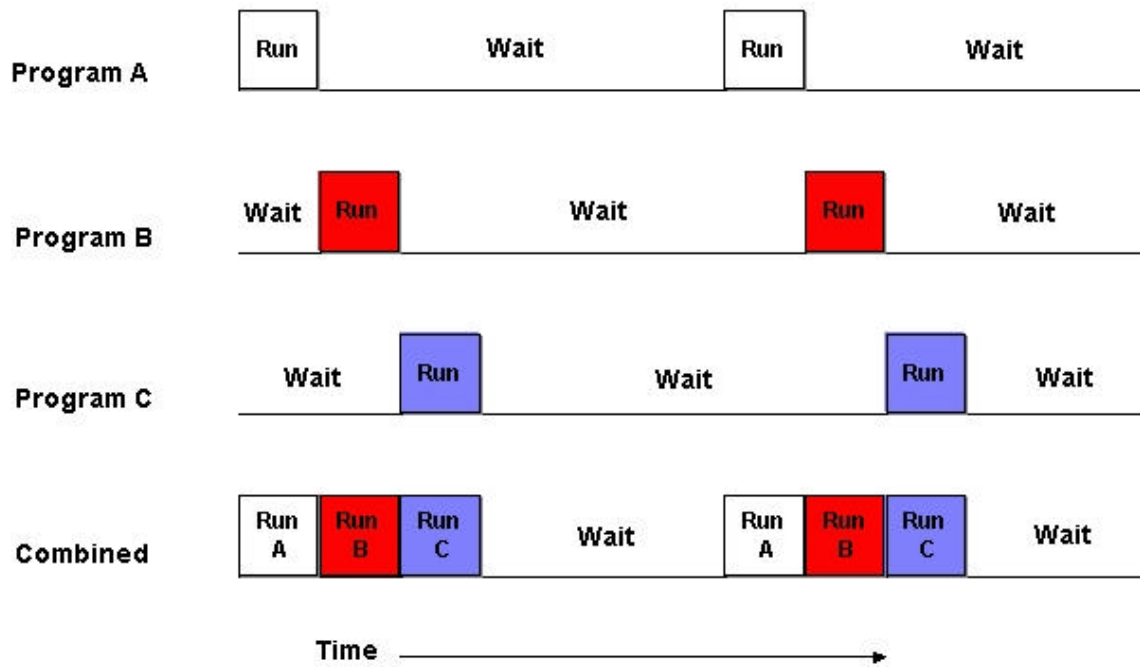
Software Structure of Monitor



⁵² William Stallings, "Operating Systems", New Jersey: Prentice-Hall, Inc., 1997. (Page 51)

Figure - 3⁵³

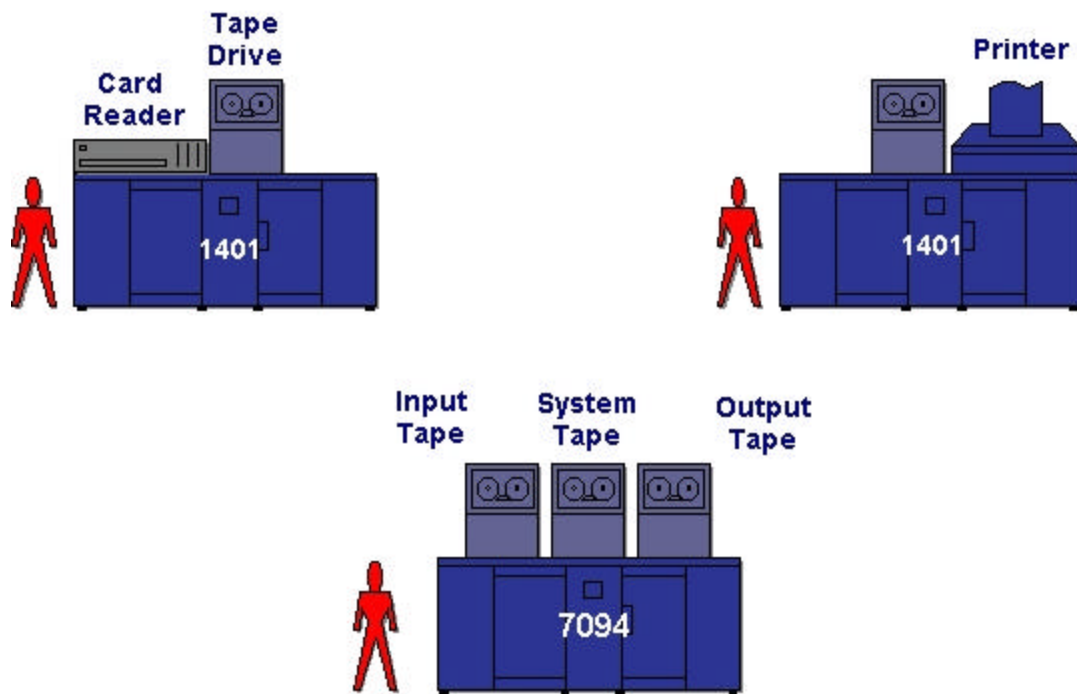
The Process of Multiprogramming



⁵³ William Stallings, "Operating Systems", New Jersey: Prentice-Hall, Inc., 1997. (Page 55)

Figure - 4⁵⁴

An Early Batch System Taks Procedure



⁵⁴ Tanenbaum, Andrew S., "Modern Operating Systems", New Jersey: Prentice-Hall, Inc., 2001. (Page 8)

Tables:

Table-1⁵⁵

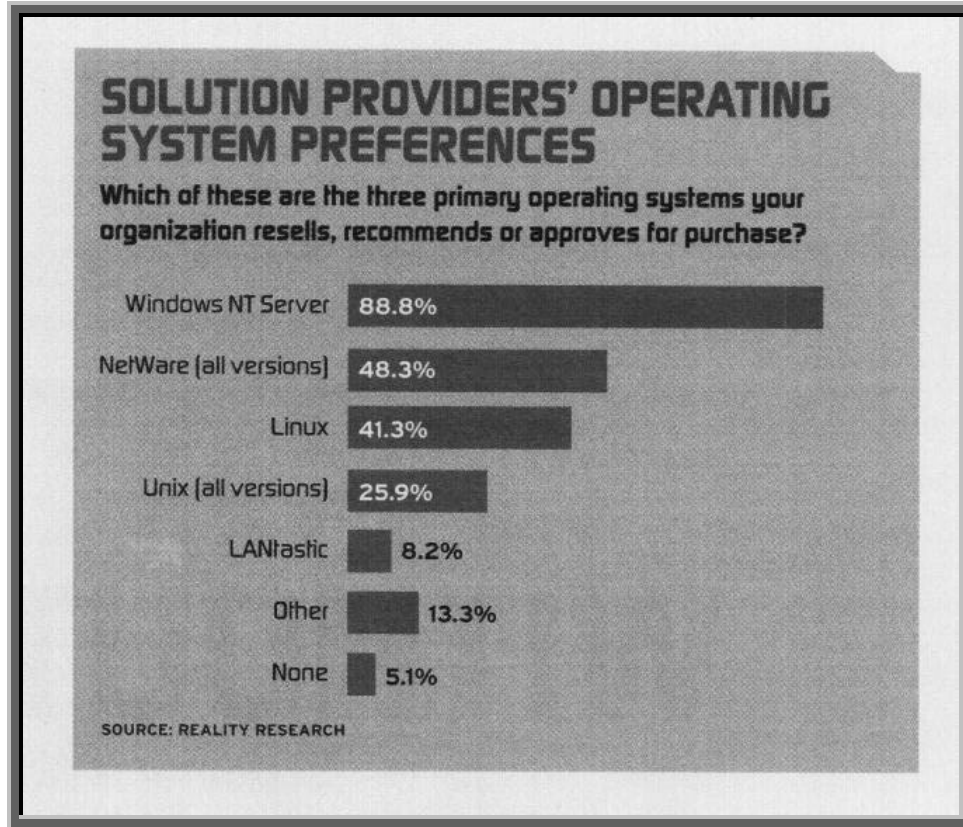


Table-2⁵⁶

Comparison of Multiprogramming and Time Sharing		
	Batch Multiprogramming	Time Sharing
Principle Objective	Maximize processor use	Minimize response time
Source of instructions to operating system	Job control language instructions provided with the job	Commands entered at the terminal

⁵⁵ Rooney, Paula, "Operating without liability", *Computer Reseller News*, Issue 930, (Jan 29, 2001), Page 14.

⁵⁶ William Stallings, "*Operating Systems*", New Jersey: Prentice-Hall, Inc., 1997. (Page 46).

6 - References:

1. Tanenbaum, Andrew S., “*Modern Operating Systems*”, New Jersey: Prentice-Hall, Inc., 2001.
2. Walters, E. Garrison, “*The Essential Guide to Computing: The Story of Information Technology*”, New Jersey: Prentice-Hall, Inc., 2001.
3. Silberschatz, Avi, Galvin, Peter Baer, Gagne, Greg, “*Operating System Concepts*” John Wiley & Sons, Inc., 2001.
4. Stallings, William, “*Operating Systems*”, New Jersey: Prentice-Hall, Inc., 1997.
5. Rosen, Saul, “*Programming Systems And Languages*”, New York: McGraw-Hill Book Company, 1967.
6. Rooney, Paula, “Operating without liability”, *Computer Reseller News*, Issue 930, (Jan 29, 2001), Page 14.
7. Kuo, Joshua, “Operating Systems- Work Ready Linux—Easy management, solid security top benefits of four Linux distributions”, *Internetweek*, Issue 877, (Sep 10, 2001), Page 23.
8. Bartholomew, Doug, “And it keeps on computing”, *Industry Week*, Issue 13, (Oct 2001), Page 73.
9. A G Ganek, E H Sussenguth, “Turning points in systems architecture”, *IBM Systems Journal*, Volume 38, Issue 2/3, (1999), Page 335-339.
10. Cummings, Joanne, “Study finds AS/400 to be cost-effective net platform”, *Network World*, Volume 10, Issue 11, (Mar 15, 1993), Page 25.
11. Johnson, Maryfan, “DEC Flip-Flops on Unix Workstation Strategy”, *Computerworld*, Volume 26, Issue 24, (Jun 15, 1992), Page 1.
12. Sigler, Lee, “Unix’s Infancy: An Operating System Grows at Bell Labs”, *Computerworld*, Volume 18, Issue 39, (Sep 26, 1984), Page 75.
13. SystemLogic.Net. (2000). Evolution of Operating Systems. [On-Line]. Available: <http://www.systemlogic.net/articles/00/11/os/>
14. Gordon College. (2000). A Brief History of Computer Operating Systems. [On-Line]. Available: <http://www.math-cs.gordon.edu/courses/cs322/lectures/>
15. Anonymous, (2001). Timeline of Operating Systems. [Online]. Available: <http://www.osdata.com>
16. Anonymous, (2001). Operating Systems’ Quick History. [Online]. Available: <http://tsiolkovsky.cis.hcc.cc.il.us:100/cbrown/Fall%201999/CTEC%20150-01/Lecture%20Notes/Lecture%201/Default.htm>

17. Anonymous, (2001). Computer Timeline. [Online]. Available: <http://scils.rutgers.edu/~net99/comptime.html>
18. Mooney, Jim, (2001). Early Operating Systems. [Online]. Available: <http://www.csee.wvu.edu/~jdm/classes/cs258/cs258.html>
19. Professor Smith, (August 28, 2000). [Online]. Available: http://www-inst.eecs.berkeley.edu/~cs162/Students/Lecture_Notes_00.f/08-28-00_1.txt
20. Yu, Byunggu, (2001). [Online]. Available: <http://www.cs.uwyo.edu/~yu/COSC4740-01/Lecture1.pdf>