History and development of operating systems assignment

“ Operating Systems have earned the reputation for being the most critical software in a computer system “. The history of operating systems (SO) 2. The function of SO 3. The reason why SO are considered to be so critical History of Operating Systems At first, computers lacked any form of operating system. The computer user had sole use of the machine. The user would arrive at the machine armed with his or her program and data, often on punched paper tape. The program would be loaded into the machine, and then the machine set to work, until the program stopped. Later, computers came with libraries of support code which were linked to the user’s program to assist in operations such as input and output.

This would become the start of the modern-day operating system. However, these machines still only ran a single Job at a time. Originally, each user wrote all of the code necessary to implement a particular application, including the highly detailed machine level input/output instructions. Very quickly, this input/output coding needed to implement basic functions was consolidated into an input/output control system (SOCIO). Users wishing to perform input/output operations no longer had to code the instructions directly.

Instead, they used SOCIO routines to do the real work. This greatly simplified and sped up the coding process. The implementation of input/output control system may have been the beginning of the concept of operating system. Under this system, the user has complete control over all of main storage memory and as a result, this system has been known as the single user contiguous storage allocation system. Storage is divided into a portion

holding input/output control system (SOCIO) routine, a portion holding the user’s program and an unused portion.

The first single-user real storage systems were dedicated to one Job for more than he Jobs execution time. Job’s generally required a lot of setup time during which the operating system loaded, tapes and disk packs were mounted, appropriate forms were placed in the printer and time cards been punched in. When the Jobs were completed, they required considerable time to teardrop, as tapes and disk packs were removed, time cards were punched out etc…. During the Job setup and Job teardrop, the computer sat idle.

Computer users soon realized that they could cut down the amount of time wasted between the Jobs, if they could automate the Job-to-Job transition. The first major such system, which was considered by many to be the first operating system, was designed by the General Motors Research Laboratories for their IBM 701 mainframe beginning in early 1956 . Its success helped establish batch computing – the groupings of the Jobs into a single deck of cards, separated by control cards that instructed computers about the various specification of the Job.

The programming language that the control cards used was called Job control language COOL). The Era of Timesharing and Multiprogramming sass: The systems of the sass were also batch processing systems but they were able to as observed by the operating system designers that when one Job was waiting for an input/output operation to be completed before the Job could continue using the processor, some other could use the idle processor.

They realized that running a mixture of diverse Jobs appeared to be the best way to optimize computer utilization.

The process by which they do so is called multiprogramming. To take maximum advantage of multiprogramming, it is necessary for several Jobs to reside in the computers main storage at once. Then, when one Job requests input/output, the CPU abbey immediately switched to another, and may do calculations without delay. One of the major developments was timesharing system which enabled many users to share computer resources simultaneously. In timesharing mode, the computer spends a fixed amount of time on one program before proceeding to another.

Timesharing systems helped facilitate the software development process significantly. With turnaround time reduced to mere minutes, a person writing a new program will not have to wait hours or days to correct errors. With timesharing, a programmer could enter a program, compile it, receive a list of syntax errors, correct hem immediately and re-execute this cycle until the program is free of syntax errors thereby reducing development time significantly. The personal computer era Early personal computer operating systems were very diverse.

Each vendor was producing one or more operating systems specific to their particular hardware. Nearly every operating system could have radically different models of commands, operating procedures, and such facilities as debugging aids. It was the development of microprocessors made inexpensive computing available for the small business and the hobbyist, which in turn led to the widespread use of interchangeable hardware

components using a common interconnection (such as the S-OHIO, Apple II, AS-50, SIS and PC buses), and an increasing need for operating systems to control them.

The most important of the early JOSS on these machines was Digital Research’s CUP/M-80 for the 8080 / 8085 / Z-80 JPL’s. It was based on several Digital Equipment Corporation operating systems, mostly for the PDP-11 architecture. MS-DOS (or PC-DOS when supplied by MM) was based originally on CUP/M-80. Each of these machines had a small boot program in ROOM which loaded the SO itself from disk. The BIOS on the IBM-PC class machines was an extension of this idea and has more functions and features in the 20 years since the first IBM-PC was introduced in 1981.

The decreasing cost of display equipment and processors made it practical to provide graphical use interfaces for many operating systems, such as the generic X Window, System that is provided with many UNIX systems, or other graphical systems such as Microsoft Window, the Radio Shack Color Computer’s SO-9 Level II, Apple’s Mac SO, or even Vim’s SO/2. The original GUI was developed at Xerox Palo Alto vendors. The significant types of Operating Systems

Microsoft Windows has become the most widely used operating system for personal computers but there have been many others which have made a significant impact upon the development of the personal computer. CUP/M : Considered to be the ‘ first’ operating system, CUP/M was developed by Gary Killed DOS : A text based operating system developed into MS-DOS by Microsoft to run on IBM machines. MS-DOS (originally COOS) is alleged by

Gary Killed to be based upon his CUP/M operating system which IBM attempted, unsuccessfully, to buy from him, before they asked Bill Gates to create an operating system for their computers.

SO/2 : Released by IBM in 1987, SO/2 was perhaps the first real multitasking operating system. It was designed by IBM and the code written at Microsoft. Although SO/2 was expected to outsell and eventually replace MS-DOS, its actual sales figures were very poor perhaps due to an increase in memory costs at the time. LINUX : An open-source operating system developed by Lines Dorsal. Linux was developed originally for use on home PC’s but has grown to find homes on Power, Macintosh, Amiga, Atari, DECK Alpha, Sun Spars, ARM, and many other computer platforms.

Linux offers a number of different Graphical User Interfaces and can be et to look like a Windows or Mac operating system. It has been praised for its stability and speed and is, in a relatively small way, offering an alternative to the Windows operating system for PC’s. MAC SO : The Apple operating system developed to run on Macintosh machines. The Mac operating system is unique to Apple computers and yet is the second most widely used after Windows. Macintosh computers have often been favored by computer users working in graphical design fields.

Apple and Microsoft have fought over the available market for operating systems with IBM machines often considered s mainly Windows machines. Apple have lost out in the past but are regaining customers with their innovative approach to computer design and the perceived reliability of Mac SO when compared to Windows. Windows : Now the predominant operating

system for personal computers, Windows offers a Graphical User Interface based upon a ‘ desktop’ metaphor. Windows has also enabled applications to perform in a consistent manner which means that menu options look similar from one package to another.

The Functions of Operation Systems Systems makes the Hardware conveniently available to users, by managing the reward carefully to achieve good performance. You may consider Operating Systems to be managers of resources as it determines which computer resources will be used for solving which problem an the order in which they will be used. The three principal types of functions an operating system has are : Assignment and allocation of system resources such as input/output devices, software, central processing unit. Scheduling : This Function coordinates the Job an resources and follows certain giving Priority.

Monitoring : This function Monitors and keeps track of the activities in the computer system. It maintains logs of Job operation, notifies the computer operators of any abnormal terminations or error conditions. This function also contains security monitoring features such as any authorized attempt to access the system as well as ensures that all the security safeguards are in place. Modern operating systems often provide users and applications with a virtual machine, an interface to the underlying hardware that makes it appear as though the user is the only user of the machine and it’s hardware.

Whether the computer has one CPU or several JPL’s, it is usually the case that there are more processes than JPL’s. Therefore, the operating system is responsible for scheduling the processes on the CUP]. There is a finite

amount of memory that must be shared among the processes. The way this is done varies between different operating systems, but a commonly used mechanism is that of virtual memory. Several different processes may be trying to access a single 10 device and the operating system must manage these accesses.

This is a different issue than processes scheduling since often 10 is being performed for processes that are not currently executing. Some devices (e. G. Disks) have resources that can be shared mongo users and/or user processes. The operating system is responsible for managing and protecting these resources. Another important operating system function is providing support services for processes. These include: Support for 10 operations. File system management. Protection. Interrupts and Traps. An interrupt is a CPU event that is triggered by some external device.

The SO manages these devices. Each device has a diver which is used to communicate with the SO and the device. A trap is a CPU event that is triggered by a program. Traps are sometimes called software interrupts. They can be deliberately triggered by a special instruction, or they may be triggered by an illegal instruction or an attempt to access a restricted resource. The reason why SO are so critical The main reasons why operating systems are so critical is by the functions that it performs which I have explained in the last few pages.

It is also so critical as it provides a layer of abstraction between the user and the bare machine. Without an SO, it would be very hard and time consuming to do a lot of the Jobs on the computer that we take for granted. The users

and applications do not see the hardware erectly, but view it through the operating system. It is the operating system that lets us to communicate with the external devices. Conclusion Operating systems influence the way in which we communicate with personal computers. They have been developed to manage new technologies.

The development of the PC has seen Microsoft grow to be the largest supplier of operating systems. The need for reliable and easy-to-understand operating systems has prompted development of suitable systems to progress at a very quick pace. It is possible that a greater number of competing operating systems will become available to the nonuser in the future although this does mean that users may find it difficult to move from a machine running one operating system to a machine running something quite different.

The market for operating systems will continue to grow as the number of devices that can use them increase and it is sure to be a competitive area. Bibliography http://computer. Housework’s. Com/operating-system. HTML http://www. Stoats. Com/kind/history. HTML http://www. Contemporaneousness. Org http://en. Wisped. Org/wick/operating_system http://Vidal. Vaduz. Let/; projects/index. HTML http://faculty. Kowtows. Du/riskiest/343/notes/goddess. HTML