

CST Part IA: Operating System, SV 4  
Joe Yan  
2017-3-15

## 1 2009P2Q3

- (b) (i) Pipe is a way to pass the information from one process to another. It is a form of inter-process communication. Pipe is a one direction communication. There is a writer process sending data to a reader process. The pipe is implemented by a buffer space in the memory. The process sending data writes the data to the tail of the buffer space and the process accepting data read from the head of the buffer space until there is no more data to read and the reader process blocks. Pipe is treated as a file in Unix and so needs a system call to get two file descriptors such as read or write descriptor to corresponding process reading or writing.
- (ii) The shell is a tool to explain the entered commands to execute.  
The simplest shell run by the following instructions.  
A shell keeps doing the loop. The loop starts with waiting for a command input string. After the shell reads the input, it parses the input and fork off a new child process by the system call `fork()`. The return value of `fork()` will be stored. If the value is smaller than 0 then the shell prints the error information and repeats the loop. If the value is 0 then the process is a child process and the shell makes the child process do the work by the system call `execve()` with the parsed command input as parameter. If the value is not zero then this value is actually the PID of the created child process which means the shell should wait for the child process finishing its job by the system call `waitpid()`. Finally the shell repeats the loop.

## 2 2014P2Q4

- (a) A blocking IO means a process will wait for the IO operation to be completed. So when process starts to wait for IO, it will enter a blocking state until the CPU finishes the IO operation. Then the process will enter the runnable state and wait for the scheduler to put it on the CPU.  
A non-blocking IO means the IO will not stop process from executing. The process will check for the IO operation later.  
The OS can use a buffer space in the memory to pre-read the data from disk or Internet stream in advance. So when the IO blocking process asks for data, it can directly use the prepared data in the memory without blocking.
- (b) Use a double buffering strategy will first read the first block from the disk with blocking which takes 3 ms. After that the CPU will call DMA (hardware assumption made) to continue to fill another buffer with the second block which will take 3 ms. Then the CPU processes the first block without blocking which takes 2 ms. After finishing the process of the first block, the CPU will wait just 1 ms more for the data in the second buffer to be prepared. In a long term, the process will be 40% faster than previous single blocking buffer strategy.
- (c) The polled IO means the processor will keep asking whether the IO device is prepared for the next IO operation until the IO device is ready. So if the latency of the IO device is short then most of the computing time of the processor will be used for sending queries to the IO devices which is useless effort and wasteful.  
An interruption will be much better. The CPU invokes a IO operation to a IO devices

and the invoke will be push to the back of the operation queue in the IO device. When the IO device finish those operations with higher priority and get prepared for that job, it will send a interruption to the CPU to proceed the IO operation.

- (d) The DMA is a hardware which transfer the data between the memory and IO devices. This frees the CPU time to do other things. Also the data is directly written to the memory and this takes less effort than transferring by CPU because the CPU reads from the device and then write to the memory. And also the CPU is not directly connected to the memory. It might needs to clean the CPU cache to leave space for the incoming data to the memory. This cache clean up will not happen if the transfer is done by a DMA. The CPU first sends a command to the DMA and DMA puts the command in the queue. When DMA get the transfer job from the queue, it will write the data directly from the device to the required memory address or vice versa. After the transfer is done, the DMA will send an interruption to the CPU to tell CPU the data is prepared in the memory and the process which was asking for the data now can proceeds.
- (e) The biggest issue of a heterogeneity in IO system is that different devices has different IO standard. So the OS will be required to provide reliable interfaces for these devices which are called device driver.  
Also their latency and bandwidth may be enormously different from each other. The OS should guarantee their consistency.

### 3 Soft/Hard Link

The soft link is a file stores a string which means a path from root to another file or directory and it will be interpreted by the OS when being executed. However the hard link is an entry that directly references an index node in the file system and each hard link really gives a file a name which is stored in the directory.

Renaming a hard link means renaming a file or rewrite a name of the file in that specific entry but the entry still points to that index node so nothing actually changes in the file.

Renaming a soft link is just renaming this soft link and the path stored in the soft link is the same.

When opening a file by its soft link, the OS will first interpret the string stored in the soft link to a path to some other entry in the file system until find a hard link which is really a entry pointing to a index node of a file. If the path means nothing e.g. the hard link get referenced is moved or deleted then the soft link will be useless storing a path which means nothing. When opening a file by its hard link, the process invokes to open the process will read the file from the index node if it has the rights to read.

Writing the file really means is writing to the data stored in the file system which as many blocks directly or indirectly pointed by the index node of the file.

When deleting a soft link, nothing changes for the file. It is the same as deleting a file storing a string which means that specific path.

When deleting a hard link, the file actually loses a name or a entry from a specific directory. A file will need at least one hard link to be pointed or found in the file system. If all hard links are deleted (reference counter in the index node is 0) then the OS will really free its index node and data blocks.