Christian Boni
CS 1541
4.17.15

File/Storage System Implementation

First of all the Storage system is implemented by a class named Storage System and takes only one parameter in its constructor which determines which level of RAID to perform or to not perform any form of RAID at all. Storage System has two public methods, which are Read and Write and one private method named XOR which is used for RAID 4 and 5 that include the use of the parity sector. The Write method takes two parameters, a sector number and a byte array of data. Write uses the given sector number and stores the byte array in that particular sector file in a disk folder. Read only takes one parameter which is the sector number and returns a byte array of data located at that sector file in the disk folder. The XOR method simply takes two byte arrays of data and uses a for-loop to XOR each byte at each index of the two arrays together. The XOR method then returns the resulting byte array of this XOR.
NOTE: If program does not run on Windows, replace all of the forward slashes (/) in the any path Storage System class to double backslashes. (\\) Double needed for Java escape codes.

Now moving onto the File system implementation that is done through the File System class. Similarly, it takes only one parameter in its constructor which determines the RAID level or no RAID at all. Two separate classes that primarily help the File System class are Sector and Node. The Sector class is the actual object that is written out to disk as an array of bytes. Sector takes an integer as its only parameter and this integer is stored as its own sector number. However, the root sector is a special case and uses another parameter that is a linked list of the free sectors on disk. The Sector class also has one more form of instance data, which is an array list of nodes that are denoted as its children. A sector's children can either be a directory or a file. Now the Node class simply contains its own name and an array list of sectors numbers where its data is stored (this is only the case for files, directories only are stored at one sector location). Looking back at the File System class, it has its own Read and Write methods which are almost identical to those of the Storage System. They differ only in the way that they use Sector objects instead of byte arrays. Looking ahead this conversion id done by then relying on two more methods that of which are Serialize and De-Serialize. These two methods do the work of converting Sector objects to byte arrays before they are written to the Storage System and also by converting byte arrays to Sector objects as they are read in from the Storage System.

Raid 1 Implementation

Editing the existing Storage System class allowed for the RAID 1 implementation. The Write method was simply adjusted to write the same sector file to 2 separate disk folders: Disk and Disk_Mirror. Now the Read method was similarly slightly tweaked to read from the mirrored disk in case of a missing (deleted) sector. Noting that Raid 1 cannot recover from the same missing sector on both the disk and mirrored disk.

Raid 0 Implementation

Once again we can edit the existing Storage System class further to allow for the RAID 0 implementation. This form of RAID now uses three disks, simply named Disk1, Disk2, and Disk3. Now the Write method adopts modular division of the given sector number by 3 to determine which disk folder the sector file will be written to. Similarly, the Read method also uses the sector number's modular division by 3 to determine which disk folder to read the sector file from. Finally recall that Raid 0 cannot recover from any missing sector on any of the disks.

Raid 4 Implementation

Similarly to RAID 0 this form of RAID uses three disks, named Disk1, Disk2, and Disk3. However, now the Write method now uses modular division of the given sector number by 2 (instead of 3) to determine which disk folder the sector file will be written to. This modular division is done by using two now since the 3rd disk is reserved for parity only. Every time a sector is written to either Disk1 or Disk2 its given sector pair is then read in from the disk and the XOR method is used on the pair to get the resulting parity sector to be written to Disk3. Now if a given sector does not currently have a pair then the parity sector must contain the same data as the sector itself. Similarly, the Read method also uses the sector number's modular division by 2 to determine which disk folder to read the sector file from. Furthermore, in the case of missing sector the read method will now use the XOR function in combination with the missing sector's pair to get the data from the missing sector. Finally, remember that RAID 4 cannot recover if 2 out of 3 sectors of a sector pair and its parity are missing.

Raid 5 Implementation

Similarly to RAID 0 and RAID 4 this form of RAID uses three disks, named Disk1, Disk2, and Disk3. Additionally this form of RAID is almost identical to RAID 4 despite the parity sectors being distributed and not being limited to only Disk3. The Write method now uses integer division of the given sector number by 2 then takes this value modular 3 to determine which disk folder the parity sector file will be written to. Identically to RAID 4 modular division of the given sector number by 2 is

used to determine which disk folder the sector file will be written to.   This new algorithm combining integer and modular division is done since the 3rd disk is not only reserved for parity anymore. As with RAID 4 every time a sector is written to disk its given sector pair is then read in from the disk and the XOR method is used on the pair to get the resulting parity sector to be written back to disk. Now if a given sector does not currently have a pair then the parity sector must contain the same data as the sector itself. Similarly, the Read method also uses integer division of the given sector number by 2 then takes value modular 3 to determine which disk folder the parity sector file will be read from; and still uses modular division of the given sector number by 2 to determine which disk folder the sector file will be read from. Furthermore, in the case of missing sector the read method still will use the XOR function in combination with the missing sector's pair to get the data from the missing sector. Finally, remember that as with RAID 4, RAID 5 cannot recover if 2 out of 3 sectors of a sector pair and its parity are missing.