

XFS
eXperimental File System
Version 1.0

Dr. K. Muralikrishnan
`kmurali@nitc.ac.in`
NIT Calicut

October 24, 2012

Contents

1 Introduction

XFS or Experimental File System is a file system architecture designed for XOS (Experimental Operating System). XFS is a simple filesystem which has no directory structure.

XFS spans the entire disk in XSM (Experimental String Machine). The disk consists of a linear sequence of 512 blocks. The basic unit of disk in XSM is a **block**. The size of the block is equal to that of page in memory (512 words). The total capacity of the disk is $512 \times 512 = \mathbf{262144}$ words.

Any particular block in the disk is addressed by the corresponding number in the sequence 0 to 511 known as the *block number*.

2 Disk Organization

The disk is organized by the file system as shown below.

Block No	Contents
0	OS Startup Code
1	Page Fault Handler
2	Timer Interrupt Handler
3	Interrupt 1 Handler
\vdots	\vdots
9	Interrupt 7 Handler
10	File Allocation Table (FAT)
11	Disk Free List
12	Unallocated
13 – 15	INIT Code
16 – 447	User Blocks
448 – 511	Swap Area

Figure 1: Structure of the disk

- **OS Startup Code:** Block 0 is the location of operating system code required during machine boot. (Refer ...)
- **Exception and Interrupt Handler:** Blocks from 1 - 9 is intended for ISR (Interrupt Service Routines) for interrupt and exception handling which is done by the Operating System. (Refer ...)

- **FAT** or File Allocation table contains details about the files stored on the disk. (Refer ...)
- **Disk Free List** has 512 entries, one entry for each block in the disk, indicating whether it is used or unused. (Refer ...)
- **INIT Code** It has the code for INIT process, which is the first user program run by the OS after startup. (Refer ...)
- **User Blocks.** Blocks from 16 to 447 are used to store user data files and user programs. (Refer ...)
- **Swap Area** The file system also provides a swap area for the operating system to implement demand paging. Users cannot directly store data and programs in the swap area.

3 File

A file is a collection of blocks identified by a name. Every file in the disk has a *Basic Block* and several *Data Blocks*. They are defined as follows:

- **Data Blocks :** These blocks contain the actual data of a file.
- **Basic Block :** It consists of information about the data in a file.

Index	0–255	256–511
Content	Block List	Header

Figure 2: Structure of the basic block of a file

- **Block List :** It contains block addresses of all data blocks in the file.
 - * The block list consists of 256 entries.
 - * Each entry is of size one word.
 - * The value contained in an entry of the block list gives the block number of the corresponding data block in the disk. All invalid entries are marked with 0.
- **Header :** The header contains the meta information relating to the file. The header fields is 256 words long. It can be used to store details like permissions, ownership etc related to a file, similar to inode in Linux and UNIX operating systems. However XOS doesn't use the header field.

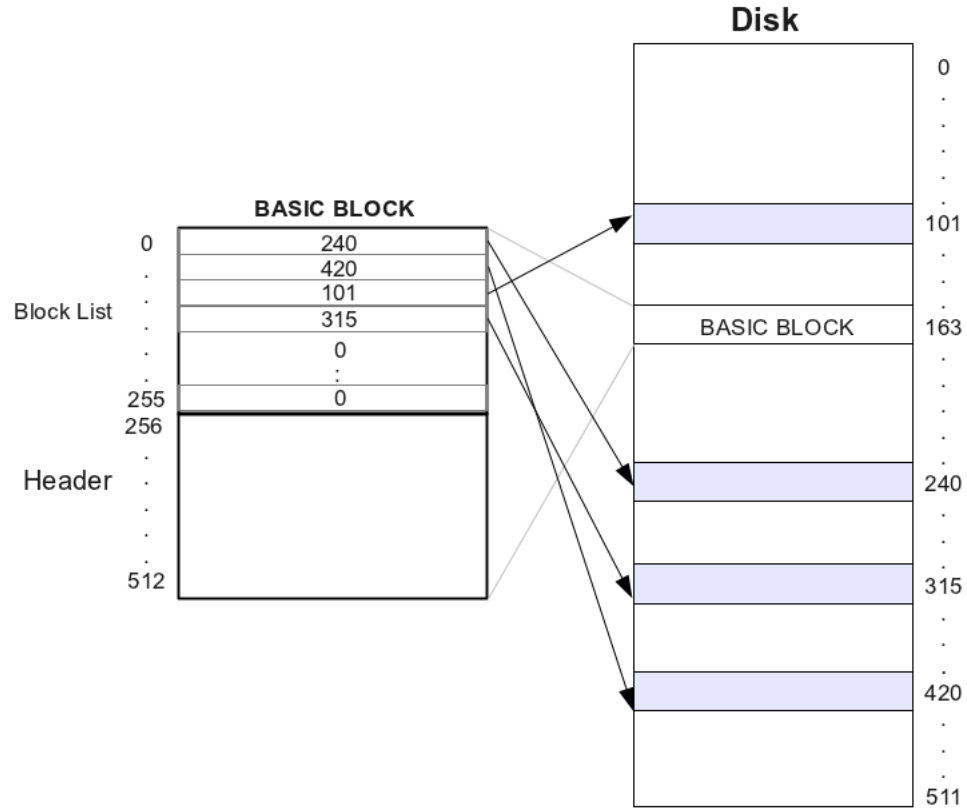


Figure 3: Example illustrating the basic block of a file

Example 3.1 Consider the example illustrated by figure ???. From the figure, we infer the following.

- The zeroth data block of the file resides in the disk in block number 240.
- The first data block of the file resides in the disk in block number 420.
- The second data block of the file resides in the disk in block number 101
- The third data block of the file resides in the disk in block number 315
- Rest of entries are empty and are marked as 0

3.1 File Types

There are two types of files in the XSM architecture. They are:

3.1.1 Data files

These files contain data or information that is used by the programs. They can occupy a maximum of 257 blocks (1 basic block + maximum 256 data blocks). Data files have an extension `.dat` in the filename.

3.1.2 Executable files

These contain programs that the user wishes to run on the machine. They occupy 4 blocks (1 basic block + 3 data blocks) of the disk. Executable files have an extension `.xsm` in the filename.

4 File Allocation Table (FAT)

File allocation table (FAT) is a table that has an entry for each file present in the disk.

- FAT of the filesystem consists of 64 entries. Thus there can be a maximum of 64 files.
- Each entry is of size 8 words.
- Total size of the FAT is thus 512 words, which occupies 1 block.
- It is a disk data structure and occupies block number 10 on the disk.

4.1 Structure of FAT

The structure of a FAT entry is shown below

0	1	2	3 – 7
File Name	File Size	Block no: of basic block	... Free ...

Figure 4: Structure of a FAT entry

The FAT entry consists of the

1. **File Name** : It is an identification of a file. It can be of maximum 15 characters (and thus requires 1 word). Typical file names are `student.dat`, `calc.xsm`.

2. **File size :** It indicates the number of words occupied by a file. A file will contain minimum of 1 block (basic block), and a number of data blocks. Thus, the size of the file can vary from 512 words to $512 \times (256 + 1)$ words. It occupies one word in the FAT entry.
3. **Block number of basic block :** It contains the block number where the basic block of a file resides in the disk. It occupies one word in the FAT entry.

5 Disk Free List

The Disk Free List is a data structure used for keeping tracking of unused blocks in the disk.

- The Free List of the disk consists of 512 entries. Each entry is of size one word.
- The size of the free list is thus 1 block or 512 words.
- It is present in blocks 11 of the disk. Refer figure ??.
- For each block in the disk there is an entry in the Disk Free List which contains a value of either 0 or 1 indicating whether the corresponding block in the disk is free or used respectively. Each entry in the Disk Free List is of size one word.
- Blocks 0 to 12 are system reserved and are marked as 1 in the Disk Free List so that they cannot be used for saving user files. The file system also ensures that user files are not stored on the Swap Area.