

Baby Cow File System

Spec 1

2005/09/05

Design principles:

1. Assume CHS are 24 bits long, so each link is 3 bytes long, only max 8 GB hard disk is supported. Really don't know how to control a larger hard disk by using port : (
2. Stable is the most important, second is speed. So my file system is atomic, either the operation is fully complete, or it is not. There is an atomic field, which is 4096 bytes long, when you write anything to the partition, backup the original block to that field first. !!! This will highly decrease the performance because write 1 block, it performs 2 write operations. !!! ReiserFS said they do it the same way, but they have better algorithm to fix this performance problem, if anyone know it, please tell me :)
3. My file system is not journal, but it is safety enough for single read/write operation.
4. I will not put all the field too tight, that mean even the field is 1 bit width, I will make it occupies 1 byte because I don't want to perform &/| operation to retrieve the byte, it waste the CPU clock.

Feature will be added late:

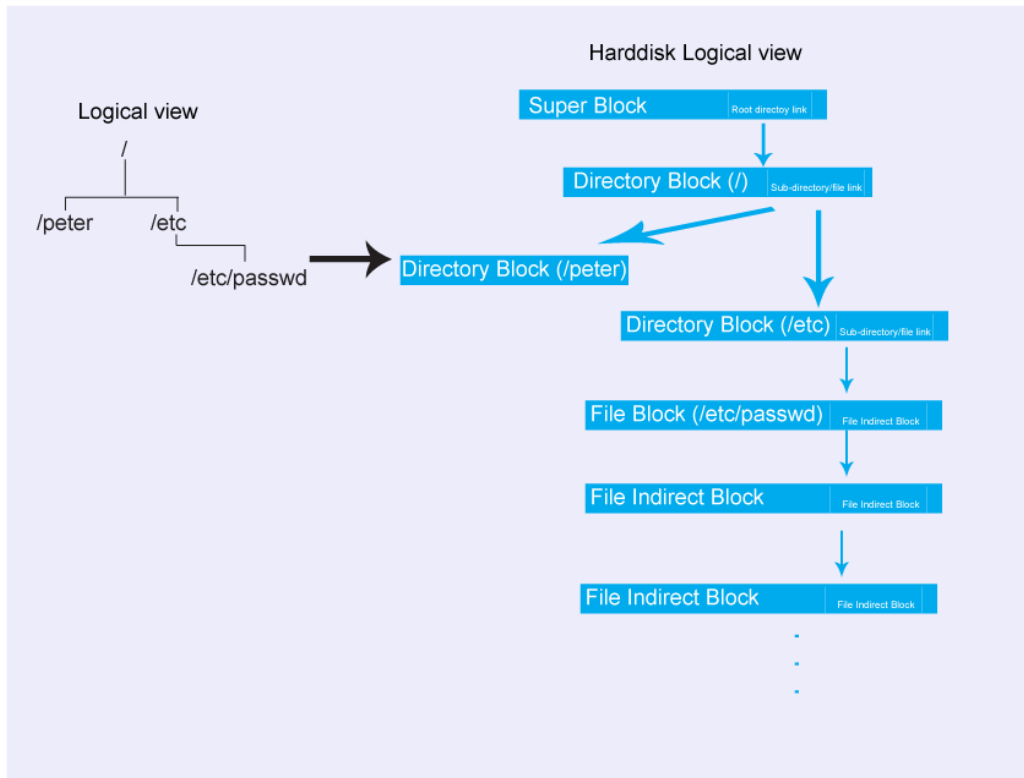
1. Map a directory of file to a memory
2. Better permission format, more suitable to my operating system.

Baby Cow file system specification

Number of file within a directory	4294967296
Number of sub directory	4294967296
Maximum file size	2^{64} / 16777216 TB
Maximum filename	500 single byte character
Maximum directory name	500 single byte character
Maximum partition size	2^{42-4} *Block size – 4096 Bytes
Block size	Any but must larger than 4096 Bytes

Hard disk layout

Super block	Atomic field	Any other block
4096 Bytes	4096 Bytes	Excess bytes in the partition



Super Block: fixed 4096 Bytes long, used to store the information of the partition. It must be located in the first 4096 bytes of the partition.

Byte Offset	0	2	502	510	518	526	528	536
Name	ID	Partition Name	Root directory link	Create time	Last modified time	Block Size in KB	Number of Free address Block	Unused
Width	2	500	8	8	8	2	8	3560
Example value	'S';'B'	"partition name"	LBA	Second since	Second since AD	4 (for 4096KB)	4 (for 4 blocks)	-

				AD 2000	2000	block size)		
--	--	--	--	---------	------	-------------	--	--

Descriptions:

1. Root directory list link, link to the block of root directory of partition, 24bit CHS value.

Free Address Block: there can be many free address list block, depend on the size of the partition, but these blocks must be following the super block.

Byte Offset	0	3
Name	ID	Free Block List
Width	3	Block size - 3
Example value	'F','A','B'	Figure 1

Byte Offset	0
Width	1
Example value	0 : free 1 : used

(Figure 1)

Number of Block = (Partition Size – 4096(Super Block size)) / Block Size

Free Address Block Capacity = (Block Size – 3 bytes) blocks. One Free Address Block can arrange up to (Block Size – 3 bytes) number of block.

Number of Free Address Block = Free Address Block Capacity / Number of Block

Directory Block : used to store the information of the directory and the links to its sub-directory.

Byte Offset	0	3	503	507	511	520	528	536	544
Name	ID	Directory Name	Number of File	Number of Directory	Permission	Create time	Last modified time	Directory indirect block	File or Directory links
Width	3	500	4	4	9	8	8	8	Block size - 544
Example value	'D','I','R'	"/root/password"	52	64	rw-rw-rw	Second since AD 2000	Second since AD 2000	Block ID	Figure 2

Descriptions:

1. Sub-directory/file link : each 4-bytes represent as follow:

ID	LBA
0 (File)	Block ID
1 (Directory)	Block ID

(Figure 2)

Number of link per directory block = $(\text{Block size} - 542) / 8$

Directory Indirect Block

Byte Offset	0	3
Name	ID	File/Directory links
Width	3	Block size - 542
Example value	'D','I','B'	Figure 2

Number of link per directory indirect block = $(\text{Block size} - 3) / 8$

File Block : used to store the information and content of the file.

Byte Offset	0	4	504	513	521	529	537	545
Name	ID	Filename	Permission	Create time	Last modified time	Filesize	File indirect block link	File Content Block
Width	4	500	9	8	8	8	8	Block size - 537
Example value	'F','I','L','E'	"/root/a/b.txt"	rw-rw-rw	Second since AD 2000	Second since AD 2000	20	Block ID	Block ID

File Content Block : used to store the information and content of the file.

Byte Offset	0
Name	File Content
Width	Block size
Example value	-

File Indirect Block : used to store the link, which pointed to File Content Block

Byte Offset	0	3	Block size - 8 - 3
Name	ID	File Content Block Link	File Indirect Block Link
Width	FIB	Block Size – 8 - 3	8
Example value	'F','I','B'	-	-