

## OPTIMIZATION OF WORK PERFORMANCE WITH LINUX OPERATING SYSTEM

**Vasile Cristian**

*University of Craiova, Faculty of Agriculture  
E-mail: [cristi\\_vasile\\_4you@yahoo.com](mailto:cristi_vasile_4you@yahoo.com)*

**Abstract.** In this article there will be made a presentation as detailed as possible of the most important characteristics of Ext2 files, which are some of the most used file systems of LINUX operating system. Also here there will be analyzed the main settings from Ext2 file system, and the base attributes that characterize this type of files.

Also there will be presented the base structure of Ext2 file system, on fields and blocks, for a correct understanding of the work type and administration approach by LINUX operating system.

All these characteristics have the purpose of presenting a few aspects concerning the work approach with Ext2 files and some optimization work methods with LINUX operating system.

**Keywords:** files, folders, setting, attributes, fields, optimization, connections

### **Ext2 file system presentation**

Analyzing the gestioned file typology and work methodology with these, it can be said that Ext2 file system is one of the most utilized file systems in the activities made with LINUX operating system.

At first, the Linux operating system used a very rudimentary file system, meaning that it imposed certain limitations at the length of file name (maximum 14 characters) and also at the files size (maximum 64 MB).

Therefore, experts in the field have always tried to improve how can work with Linux operating system. Thus, a first variant of the new files system who was implemented is the Ext, which brought much higher performance for Linux users, meaning that allow working with files whose name was 255 characters and up to 2GB in size.

However, the specialists has aimed at continuous improvement the performances of the Linux operating system and therefore has been implemented the Ext2 files system, which eliminated many of the grievances of those who operating with the Linux operating system.

Ext2 files kept the standard structure and the semantics utilized by Unix operating System, but in the same time a series of new characteristics have been implemented. So, due to the inclusion of kernel optimization, Ext2 file operating system is much more robust and is characterized by high performance during work, and the structure of these files allows future extensions as in the case of access list construction, of compressing and deleting operations.

Also, by the work model and the structure, it can be said that Ext2 file system is very similar with Berkeley Fast File System, between them existing, of course small implementation differences.

### **Ext2 file type characteristics**

As mentioned, the file system Ext2 supports the standard files of the operating system UNIX : common files, folders, special files. The names of the Ext2 file can have up to 255 characters, but there is also the possibility to extend the name up to 1012 characters. Also, by the Ext 2 system you can administrate files with big dimensions that can reach up to 4 TB, which allows a special reliability when working with files.

Another interesting characteristic of the Ext2 file system is the fact that some block of dates, approximatively 5% of their capacity is reserved only for the computer's administration or the network, which will allow restoration operations, which are necessary when the running processes have taken all the space of their file systems.

Ext2 file system has a series of extra characteristics in comparison with standard files from files of Unix:

1) Allows keeping the stage in which at a certain moment with the help of the "stage" field from the kernel code. This way, when a file system is set up as R/W, then the stage of this field is going to be "Not Clean". When the file system is submitted to an "unmount" operation and then is restarted to the RO (Read Only) mode, the stage of this camp is going to be set up to the Clean value. Thus,



one can mention that when there is boot moment of the Linux operating system, the ext2fsck file will use the information of the "stage" field to decide if is necessary to check the system. Also, when an error appears in the file system, then the field will be marked as "Erroneous" and when the ext2fsck file will detect this value, it will force the checking operation of the file system.

2) When he sets up the file system the computer or network's administrator can chose the dimension of the logical block. The typical dimensions of the logical blocks are: 1024, 2048 and 4096 octets. Lots of the advantages of using some blocks of big dimensions are obtained using the preallocation techniques given by the Ext2 file system, which causes an increase in work performance.

3) The characteristics attributes of the Ext2 files permit the users to modify the kernel behaviour when it acting on a group of files. This way, the attributes can be set in the desired mode by users at both lower and higher level (at the level of file or at o level of directory) and in the second case all the files which will be created after in that folder will inherited the attributes that were set up for that folder.

4) When creating files you can chose one of the BSD semantics of System V, each one having one of the following characteristics:

a) If you chose the BSD semantic, then the files will be created with the same group id as the parent folder, which means the folder where they were created.

b) If you work with a Ext2 file system in the System V semantic, then you may have two options:

- If the parent folder has the bite "setgid" set up to the logical value "1", then the new created files will inherit the group id from the folder where they are built in but also the folders created in that initial folder will inherit both group id and setgid group.
- If the parent folder has the bite "setgid" as a logical value "0", then the files and the folders will be created with the group id taken from the calling process

5) In the Ext2 file system you have to work with the quick technical symbolical contact, which does not use block of dates to

access the files, because the name of a file that makes a connection is not stored into a block of dates but in the structure inode. Such a work technique can save a space of memory and a semnificative growth of the system speed.

6) The ext2fsck file must perform at certain intervals a check of the operating system, and if that doesn't happen for a longer period of time, can appear some problems in the work with the files. To eliminate this trouble, Ext2 offers two methods, which can force the effectuation of some checks at a certain interval of time:

a) It keeps stored the values of two fields "Maximum value mount meter" and "Assembling meter". Thus, the value of assembling meter it increments every time when the file system is set in R/W mode, and when it reaches to the maximum value the check of the file system will be forced, regardless of his status.

b) The value of fields "the last test date" and "the highest interval between two tests" are saved, and when a period of time has past, equal with the value of second field the check of file system is forced, regardless of his status.

7) The Ext2 file system allows the realization of synchronous updates of BSD type, in the sense that there is a option for setting by the system administrator to work with this type of file system, that allows as the metadata (inoduri, bitmap blocks, blocks directory and other information of indirection) to be written synchronously on the memory suport (floppy disks, memory sticks, CD, DVD, hard disk) when they are changed. This is very useful for operators because can keep a strict record of the data used, but also has the disadvantage that causes an obvious decrease of speed work.

8) The files Ext2 have a batch of attributes that doesn't exist in Unix :

a) EXT\_2\_SECRM\_FL – means the overwrite of dates with a random content before to be released with the help of the function truncate ( ), ensuring this way that it will not be possible to be restored after it is erased.

b) EXT\_2\_IMMUTABLE\_FL – permits the deny, the erase, modify or rename of a file, that means it permits the protection of the files and it permits in this case only the

access to read this files.

c) EXT\_2\_SYNC\_FL - means the possibility of some operations of assincron writing

d) EXT\_2\_APPEND\_FL – permits writing on a file, but only in the case that some information is added at the end of this file. This attribute seated denies the delete or rename of the file, being used especially for the log files.

### Ext2 system structure

Analyzing the physical structure of the Ext2 file system, which is made of groups of data blocks (*figure 1*), one might notice that it was highly influenced by the one used in working with the BSD file system, where it is formed of groups of cylinders.

Boot Sector	Block Group No. 1	Block Group No. 2	.	Block Group No. k
-------------	-------------------	-------------------	---	-------------------

Figure 1. Ext2 file system

The use of the groups of blocks in the formation of files improves very much the reliability of the Linux operating system, because in this case the control structures are recreated in every group of blocks and that is

why it is very easy to create a system of files in case unwanted a certain data superblock had been corrupted, in which case can not be recognized by the operating system to work with them. This file structure also helps to obtain some good performances concerning work speed, through reducing the distance between the table of junctions and the data blocks, because the positions of the input-output forms are reduced in that way.

Detailing this structure of the Ext2 files, one can notice that every group of blocks or superblock comprises the following elements (*figure 2*):

a copy of the control information concerning the file system (the location of the superblock, the number of blocks and junctions in the superblock, the size of the blocks on the memory support).

the group describer, which contains internal information on the irrespective file group (the location of the block on the memory support, the number of blocks and free junctions, the number of folders in the group). The number of folders is necessary to be known in order to create a balance of allocation of the folders in that group, meaning that a new folder will be created in the group with the smallest number of folders.

Superblock	Descriptors group - filesystem	Bitmap blocks	Inode's bitmap	Inode's table	Data Blocks
------------	--------------------------------	---------------	----------------	---------------	-------------

Figure 2. Detailed structure of the Ext2 files [\[http://www.linux.com/learn/docs\]](http://www.linux.com/learn/docs)

A superblock is 1024-byte sized and contains a series of fields referring to the number of blocks and junctions used or free, the block sizes, the date of their creation, the date of the last input in the blocks, the production counters, the date of the last test, the error state, the signature of the Ext2 file type of the operating system, RESUID, RESGID.

The RESUID and the REGSID fields (where the character group “RES” means “reserved”) identifies the users and the groups of users, others than the administrator, which are allowed to use the reserved blocks in the Ext 2 file structure.

The junctions table contains the whole information referring to the files kept in an arborescent memory structure. A junction

in the Ext2 file system is memorized on 128 bytes, so the double of the number of bytes used for a junction in Unix. The file addresses were also extended from 3 to 4 bytes, in order to be able to work with disk partitions which contain more than 224 blocks, meaning 16 GB of memory.

In working with the Ext2, the structure of a junction has the following extra fields compared to the standard structure of a Unix junction:

- the date of the removal of the file.
- two access lists (ACL), for the file or for the folder.
- the fragmented address.
- the file version.

These fields of the junctions can be used by the Linux operating system for a more



safe protection of the information.

For example, if an inode refers to a device, the first block number of the inode structure will refer to the corresponding number for the referred device.

We must also add the fact that in Ext2, the folders are managed through a simple chaining, which contributes to more quickly browsing of the arborescent structure of the memory, in order to find and access a certain file.

### Methods of Linux performance optimization concerning Ext2 files

Utilizing the file system Ext2, there can be performed reading in advance, meaning that when reading a certain block is wanted, kernel's Linux code will automatically initiate reading-writing operations on several contingent blocks, which have possible connections with the block on which the reading is being performed at that moment. In this way increases the possibility that the next read block is already loaded in the hidden buffer of the memory. These operations of reading in advance are normally performed during sequential file reading, but Ext2 extends this technique to directors reading too.

Because of the information structure, the file system Ext2 can also implement more allocation optimization of the necessary memory in order to store the information from files, in order of a significant increase in work speed, aspect which is not at all neglected by the Linux operating system users.

The groups of contingent blocks are used in order to group inodes and related data, in other words the kernel's code always tries to allocate the appropriate data blocks to a file belonging to the same group as its inode. Thus, when new data is written in a file, Ext2 preallocate a maximum number of 8 adjacent blocks for a new one. Preallocation coincidence rate is about 75% even for very large file systems, which induces very good writing performance even in large file systems, which shows the reliability of the Ext2 file system and allows to the Linux operating system users to work with a lot of data and information.

We must also remind another

optimization determined by the fact that allocation of contingent blocks to new files will induce an increase of sequential reading performance, later from these files, no matter their size.

### Conclusions

There can be noticed in this analysis that Ext2, by its characteristics it ensure a distinctive reliability during the performed activities, an increased working speed, special techniques for memory allocation and therefore it's one of the most popular file system utilized by Linux operation system.

Because of the fact that it operates with all standard file types and because of the fact it allows even very large files administration, up to 4 TB, the file system Ext2 is very indicated to be implemented in approaching Linux activities, while ensuring appropriate and secure data management and information.

Also because of the optimization that can be implemented in what data and information blocks reading is concerned, memory allocation techniques and special codes utilized in the kernel's writing, we can say that the file system Ext2 becomes very attractive to computer operators who handle the Linux operation system.

### References

1. C. Musatescu, *Sisteme de operare*, Ed. Radical **1999**
2. Vicki Stanfield, Roderick W. Smith *Administrarea sistemului Linux*, Ed. Teora, **2009**.
3. *Modern Concept in Operating System Design*, Tanenbaum, Prentice Hall, **1999**.
4. <ftp://ftp.uk.linux.org/pub/linux>
5. <http://devlinux.org/namesys>
6. <http://www.linux.com/learn/docs>
7. <http://www.linux.com/distrocentral/linux-security>

Received: September 2, 2010  
Accepted: November 25, 2010

