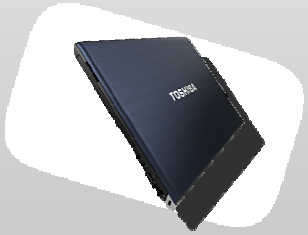
TOSHIBA
Leading Innovation



File System Performance Comparison for Recording Functions

Nobuhide Okabayashi

TOSHIBA CORPORATION
Embedded System Technology Development Dept.

TOSHIBA eco style

Toshiba Group contributes to
the sustainable future of planet Earth.

Outline

- Background
- Features of File systems
- Removal time/Fragmentation
- Measure Performance
  - basic functions
  - I/O performance
  - removal time and fragmentation
- Result
- Conclusion

Background

- Background
  - File system for embedded systems
    - “xfs” is for large volume of media and used for long time.
    - ”ext4” is standard root file system of distribution for enterprise.
    - “btrfs” supports SSD and is potentially main stream in the next generation.
  - Which file system has good performance for recording functions?

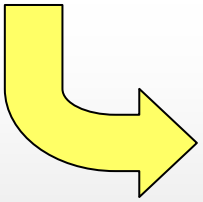
- Objective
  - Evaluate each file system in the point of needed performance for recording functions.

Features of File systems

	XFS	Ext4	Btrfs
developer	SGI	Linux developer	ORACLE
consistency of files	journal for meta data	journal checksumming	logging by COP (copy on write)
block management	extents	extents	extents
features	delayed allocation	delayed allocation backward compatibility	snapshot sub volume
merge timing	linux-2.4 linux-2.6	development version: linux-2.6.19 stable code: linux-2.6.28	development version: linux-2.6.29-rc1

Evaluated items

- Basic functions
  - processing time
    - format
    - mount
    - unmount
- I/O performance
  - write in constant rate
- File deletion
  - multiple recording may cause fragmentation.
  - removal time is related to fragmentation.



Evaluate these performance

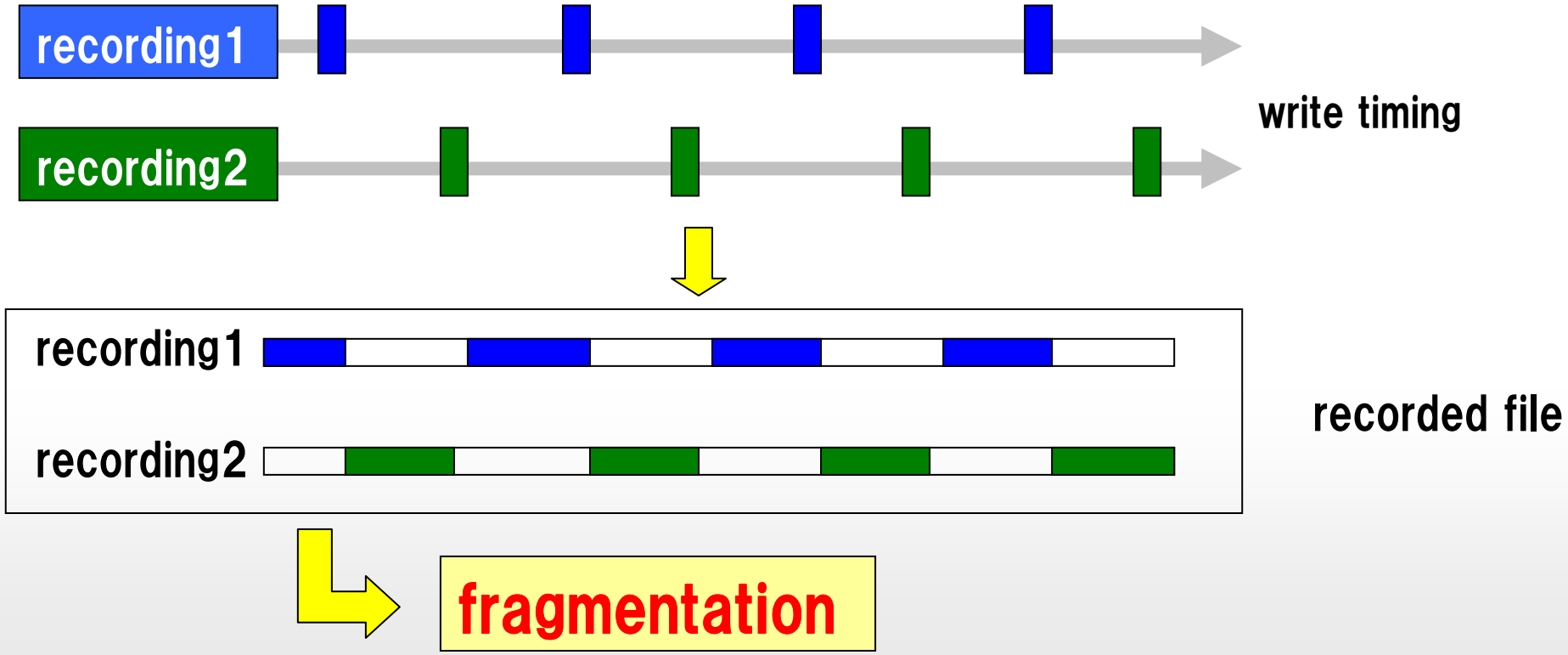
Occurrence of fragmentation

multiple recording writes files to one HDD.

write timing is switched between recordings.

file is wrote on disk with separated.

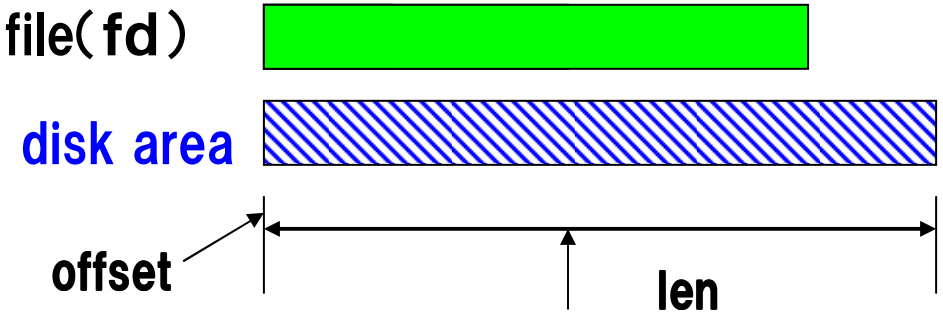
Direct I/O is used to make writing rate constant.



Reducing fragmentation

fallocate is used to preallocate blocks to a file.

→ effective to reduce fragmentation.



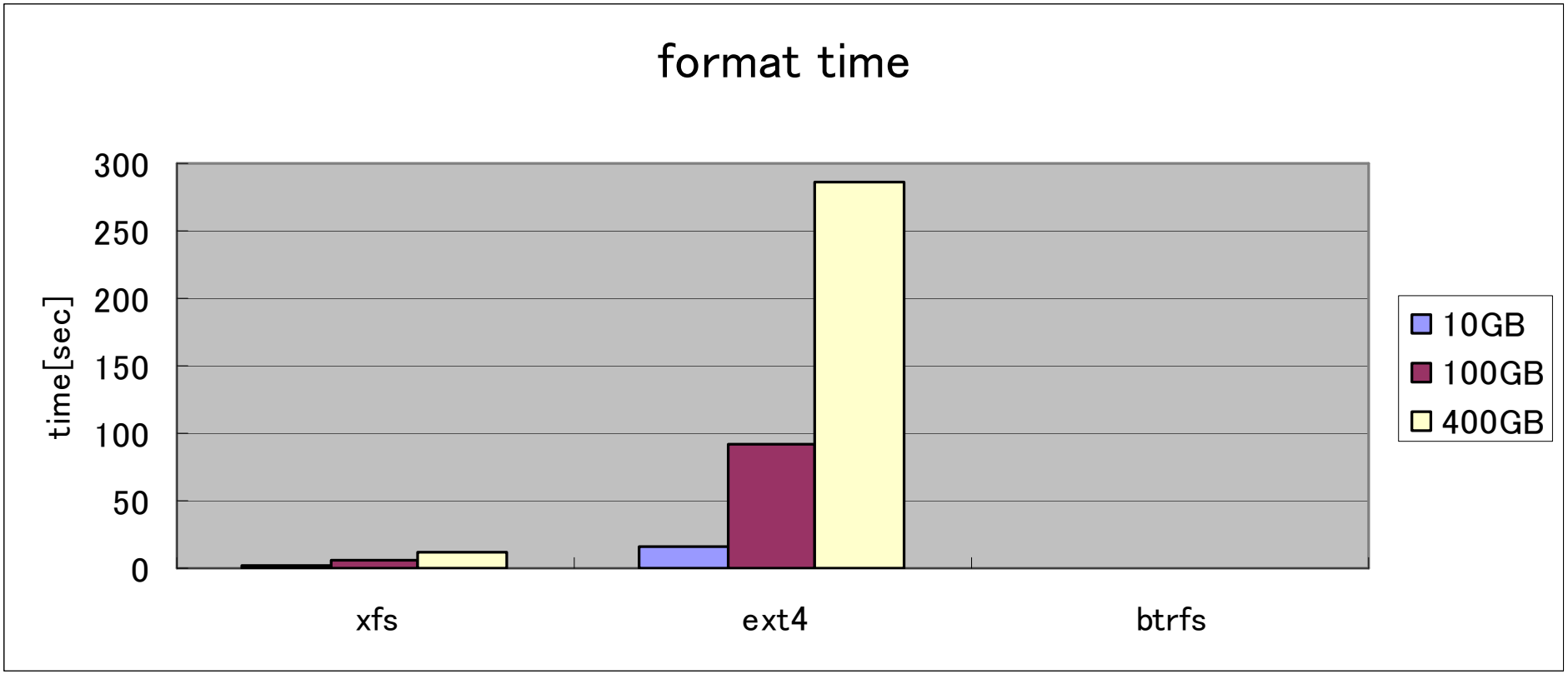
- SYNOPSIS
  - int fallocate (int *fd*, int *mode*, off_t *offset*, off_t *len*) ;
    - Specifies the beginning offset of the allocation.
    - Specifies the length of the allocation.
    - chased value of *mode* is only FALLOC_FL_KEEP_SIZE
      - Do not modify the apparent length of the file.
- available on linux-2.6.23 or later and glibc-2.10 or later.

Measure of performance

OS	Linux-2.6.39.1
CPU	ARM Cortex-A9 528MHz × 2
HDD	USB-HDD (400GB) :I-O DATA HDC-UX400
FS	xfs、ext4、btrfs

- measured items
  - basic functions
    - format, mount, unmount
  - I/O performance
    - read/reread、write/rewrite、random-read/write
  - removal time and fragmentation
    - default, fallocate (reduce fragmentation)

basic functions －format time－

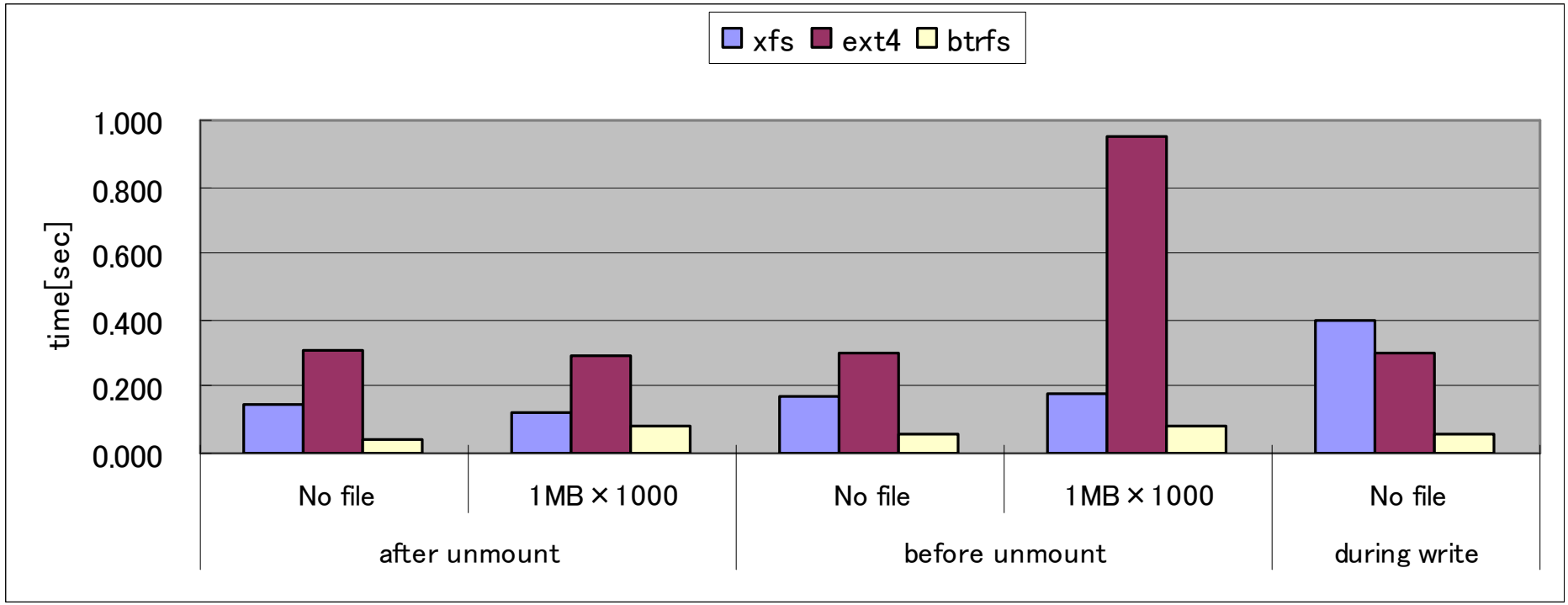


xfs,ext4: if size of partition is larger, format time gets longer.

btrfs:format time is very short (about 300ms).

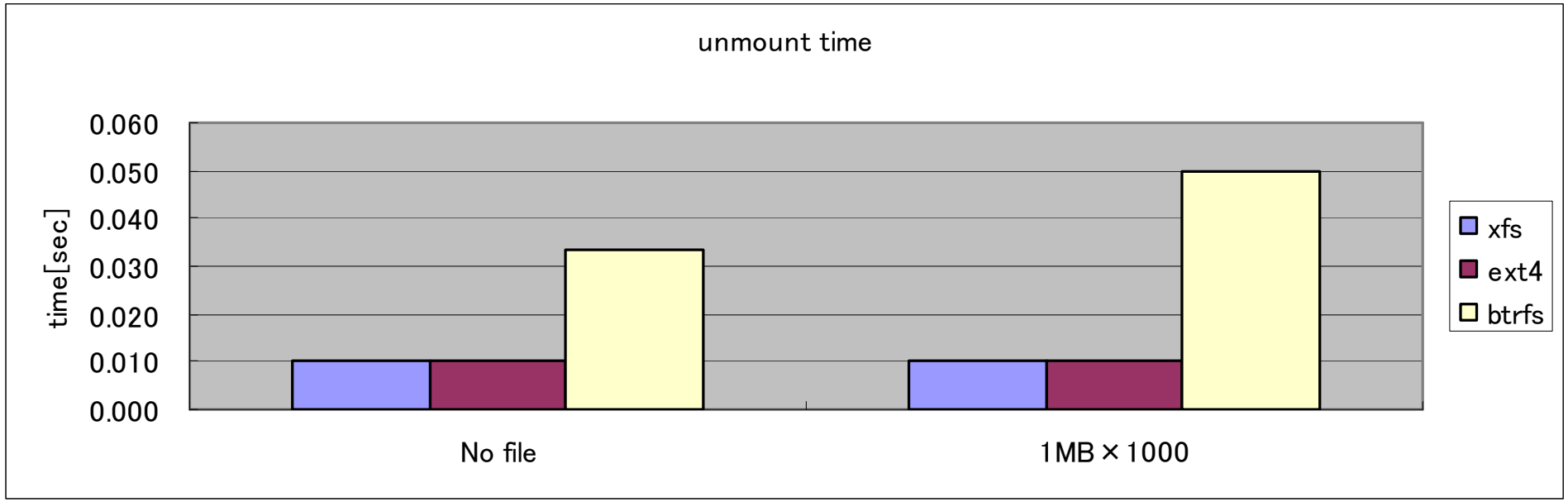
format time of ext4 is long because ext4 allocate extent in full of disk.

basic functions —mount time—



In case of ext4, recovery during mounting takes about 1 sec if there are 1000 files.

basic functions －unmount time－



There is no big differences among each file system because unmount time is less than 100 msec.

I/O performance –iozone-

- I/O performance for recording functions
  - multiple recording to one HDD.
  - Use direct I/O for writing to device in constant rate.

iozone options: # iozone –s 4G –r 564 –Ice –i 0 –i 1 –i 2 –t 4

file size	4GB
record size	564KB
thread	4
how to access	O_DIRECT
measured time	Include close () and fsync () time
measured item	write/rewrite、read/reread、random-read/write

I/O performance –iozone-

broadcast rate:about 3. 0 [MB/s]

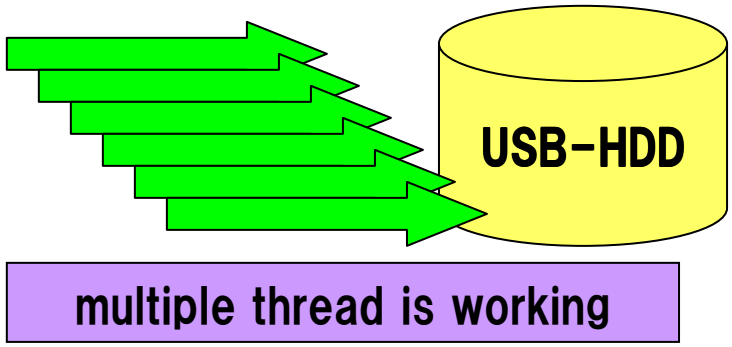
→ 4 threads:3.0×4=12.0

Satisfies performance for recording.

Removal time and Fragmentation

◆environment of creating test files

writing files in constant rate.



file size	4GB
record size	564KB
writing rate	24Mbps
thread	6

◆measure item

removal time:

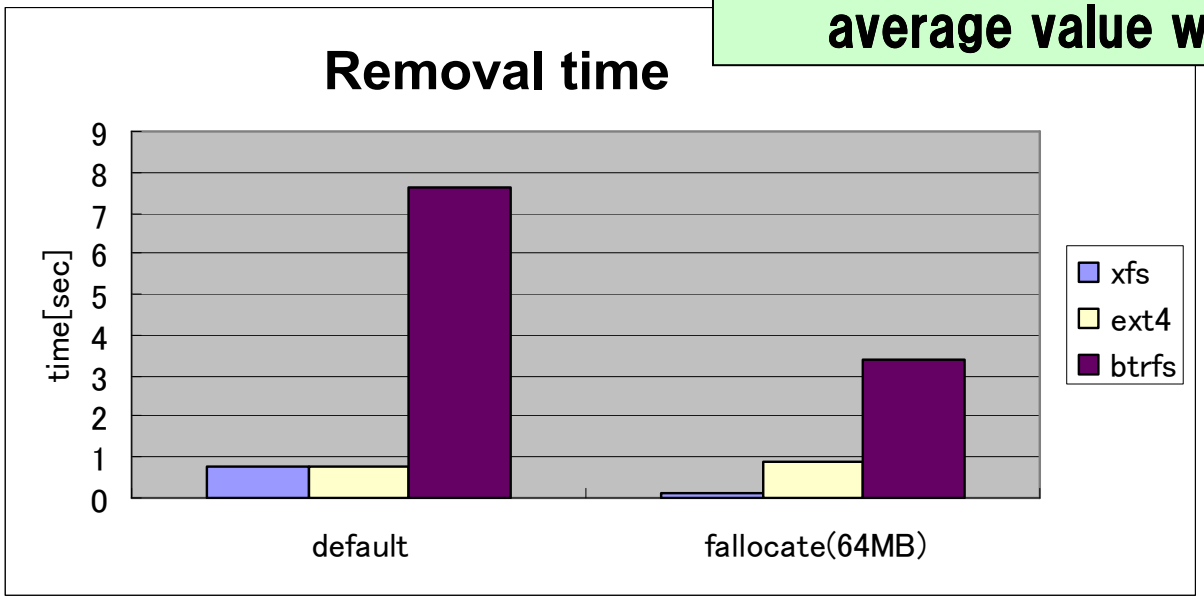
$ time rm test_file

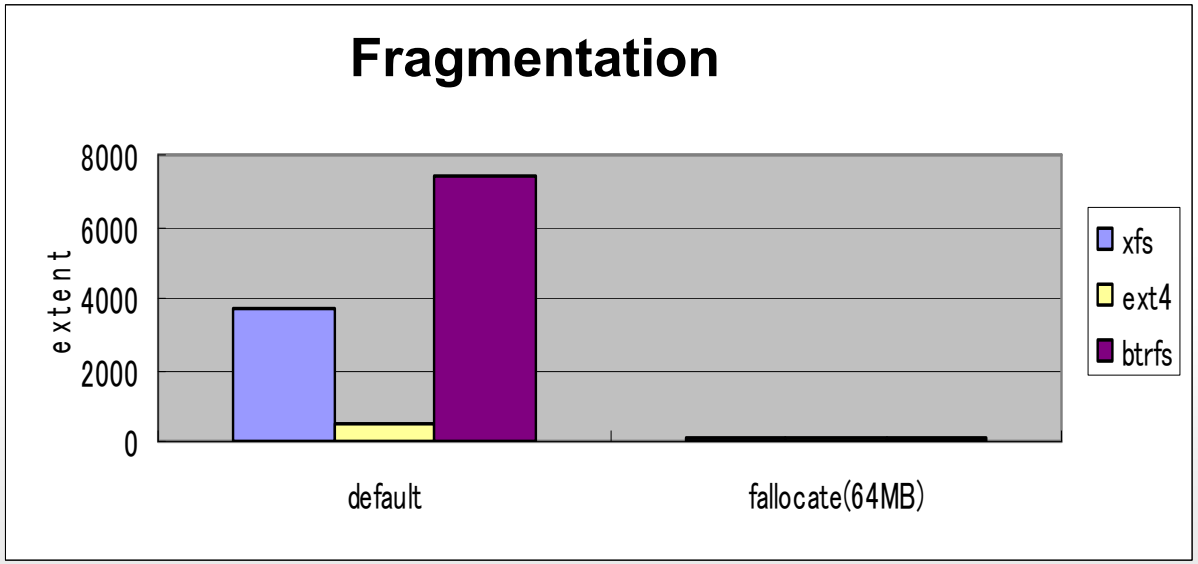
fragmentation:

measure number of extent in one test file

Removal time and Fragmentation

average value with six files





◆xfs:

Removal time gets shorter if fragmentation is reduced.

◆ext4:

Fragmentation is small.

Removal time dose not relate to fragmentation.

◆btrfs:

removal time is long.

Removal time gets shorter if fragmentation is reduced.

Conclusion

- evaluate file system performance recording functions.

		xfs	ext4	btrfs
basic functions	format	○	×	◎
	mount	○	△	○
	unmount	○	○	○
I/O performance	write	○	○	○
	read	○	○	○
removal time	time	○	◎	×
	fragmentation	△	◎	×

xfs：10 [sec]
ext4：10-300 [sec]
btrfs：0.3 [sec]

xfs：1 [sec]
ext4：1 [sec]
btrfs：3-7 [sec]

◎->very good ○->good △->Fair ×->Poor

I/O performance of each file system is enough.

performance of xfs is good balanced.

changing options, may improve performance.

Reference

1. SGI – Developer Central Open Source | XFS
   http://oss.sgi.com/projects/xfs/
2. Ext4 - Linux Kernel Newbies
   http://kernelnewbies.org/Ext4
3. Project: Btrfs
   http://oss.oracle.com/projects/btrfs/
4. Nobuhide Okabayashi:“performance comparison to XFS and evaluate configuration for recording”, The 73rd National Convention of IPSJ 6G-5 1-47 - 1-48 (2011-3)

used tool

- format
  - mkfs.xfs 2.8.18 (xfsprogs)
  - mke2fs 1.41.9 (e2fsprogs)
  - mkfs.btrfs v0.19 (btrfs-progs)
- I/O performance
  - iozone 3.397
- fragmentation
  - e2fsprogs-1.41.9
    - /misc/filefrag