TR4: Scalability Study of XFS

Xing Lin
xinglin@cs.utah.edu
University of Utah

03/08/2013

1 Introduction

This document presents the results about the performance scalability of XFS.

2 Experiment Setup

I did my experiments with a single d820 machines in the Emulab network testbed, hosted by the FLUX group at the University of Utah. An xfs was created on /dev/sdb and fio was used to generated synthetic workloads.

For scalability, I mainly looked into three aspects: IO depth (parallel writes to a single file), block size, and parallel writes to many files.

tool	version
processors	4 × 2.2 GHz 64-bit 8-Core E5-4620
Memory	128 GB DDR3 RAM
Disk	10K RPM SCSI MBF2600RC (TOSHIBA)
OS	Ubuntu12.04
fio	2.0.14

Table 1: Tools

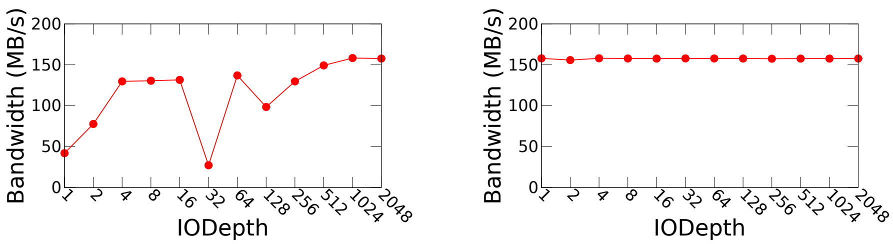
3 Results

3.1 IO Depth

I first looked into how iodepth affects the throughput of a single sequential workload. The parameters are presented in Table 2 and the results are presented in Figure 2 and 1 .

block size	duration	directio	ioengine
4 MB/4 KB	60 s	1	libaio

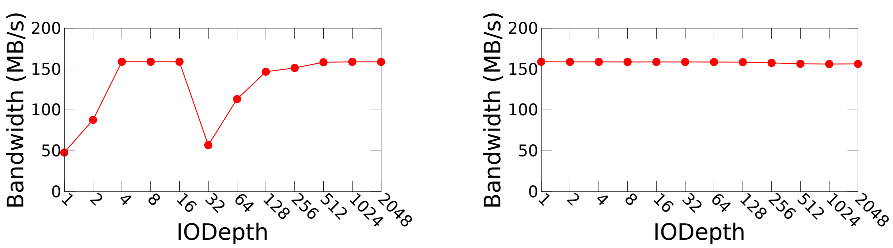
Table 2: Sequential workload parameters



(a) blocksize=4k

(b) blocksize=4m

Figure 1: The average bandwith of a SW workload, varying IO depth



(a) blocksize=4k

(b) blocksize=4m

Figure 2: The average bandwith of a SR workload, varying IO depth

3.2 Block Size

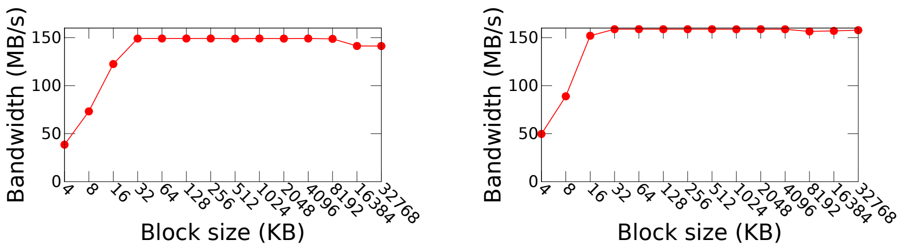
In this measurement, I fixed iodepth=1 and varied the block size. The parameters of workloads used in this set of experiments are presented in the Table 3 and results are presented in Figure 3.

3.3 Parallel Writers/Readers

In this set of experiments, iodepth was fixed at 1 and the number of files that were concurrently accessed was varied. The parameters about the workloads are presents in Table 4 and the results are presented in Figure 4 and 5.

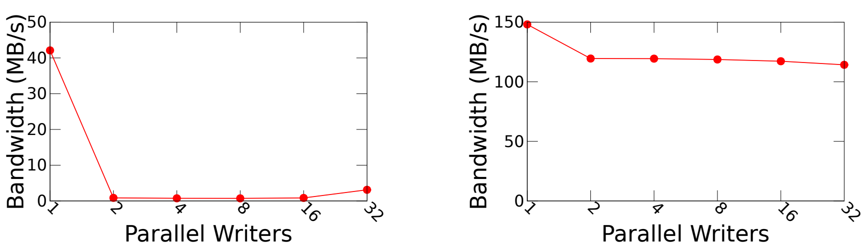
duration	directio	ioengine	iodepth
60 s	1	libaio	1

Table 3: Sequential workload parameters



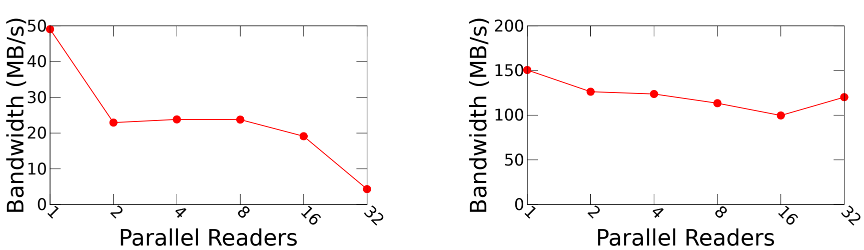
(a) Sequential Write (b) Sequential Read

Figure 3: Average throughputs of sequential workload, varying the block size



(a) blocksize=4k (b) blocksize=4m

Figure 4: Average throughputs of a SW workload



(a) blocksize=4k (b) blocksize=4m

Figure 5: Average throughputs of a SR workload

4 Discussions

Figure 1 and 2 shows bandwidths general increase as iodepth is increased for block size=4 KB while it does not for block size=4 MB. That can be explained in that as the iodepth

block size	duration	directio	ioengine	iodepth
4 MB/4 KB	60 s	1	libaio	1

Table 4: Sequential workload parameters

increases, there are greater opportunities to merge many small sequential requests into larger ones. Figure 2(a) and 1(a) shows an unexpected bandwidth drop at iodepth=32, for block size = 4 KB.

Figure 3 shows that block size starting from 32 KB can reach the maximum throughput xfs can provide for a single sequential access stream.

Figure 4 and 5 shows that as we increases the number of concurrent files to read from/write into, the aggregated throughput drops dramatically. This effect is more significant for small block sizes than large block sizes. In particular, we should avoid co-locate **sequential write workloads with small block sizes** at the same disk.

5 TODO

1. extended with random workloads