xdisk User Guide – V8.6

LINUX™ on Intel®, LINUX™ on IBM® POWER™, IBM AIX™

xdisk is a storage workload (IO) generator, able to generate multiple streaming or random IO pattern, the block size can be chosen from 512 Byte up to 64 MB; the read-write ratio can be chosen from 0% to 100%. Workload can be distributed across multiple files and multiple LINUX or AIX systems.

The output displays the IO/s and throughput (Throughput = IO/s * block size), as well as several other statistics like min, max, average IO read/write response time.

Tool and documentation can be found here:
http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS5304

Installation

Copy the platform specific binary into /usr/bin/xdisk, this pathname is required if you use the -n or -N option.

Options & Parameters

All values uses the 2^{10} calculation, so 1024 equals 1K

Mandatory parameter:

On of -S, -R, -C
-f or -F, -M, -b, -t -O

Optional parameter to specify:

-s, -i, -w, -z, -A, -x, -c, -q, -Q, -V, -n, -N

Description:

-h -? Prints help text and quits

-S[1] Sequential I/O test (file or raw device) – overwrite.
Optional argument 1 specifies to read (or write, see parameter -r) from begin to the end of the file(s) just one time, -t and –w will be ignored.
-S, -S1
-- on AIX you have to specify either S0 or S1

-R[1] Random I/O test (file or raw device) – overwrite.
Optional argument 1 specifies to read or write entire file once, -t and –w will be ignored.
-R, -R1
-- on AIX you have to specify either R0 or R1

-C size create file via sequential write; size followed by M(mega) or G(giga), range from 1 MB to 2TB.
-C100M, -C50G
If you specify -R in addition then create file will be done through random write.
-C 100G -R

-M num multiple process used to generate I/O, range 1-128, default is 1.
If the number of processes is equal or smaller than the number of files (-f or –F option), then each process gets its own file.

If the number of processes is larger than the number of files, then the processes will we distributed across the available files, each process will access its own block range.
-M8, -M10

-f name Use *name* for disk I/O, can be a file or raw device -- it is recommended to use absolute paths always.
-f /data/file1, -f /dev/mapper/vg_data-lv_data

-f name...
Use *name000, name001, name002, ...* for disk I/O. The numbers will be generated automatically, the max number is taken from the -M option. -M must be specified before -f.
This is very handy if many files are used:
-M 128 -f /data/F... ➔ /data/F000,/data/F001,.../data/F127

-f list Use *list* of files, separated by , or : or #, can be a file or raw device.
-f /data/file1:/data/file2:/data/file3

-F file >file< contains list of filenames, one per line. No blank or empty lines, no blanks before or after file name.
-F /data/list.txt :
/data/file1
/data/file2
/data/file3

-O [f:DSCWR]
Specifies how the files are accessed: either f; or one or more of "SDCRW".
f: flush the writes via fscnc()
S: opens the file with O_SYNC
D: opens the file with O_DIRECT
C: opens the file with O_CIO (AIX only)
R: opens the file with O_RSYNC (AIX only)
W: opens the file with O_DSYNC (AIX only)
If you do not use the -O flag, then reads and writes are buffered by the file system cache. If you want to measure the IO of the storage systems then you should use the O_DIRECT and/or O_SYNC flag.
-Of, -OD, -OS, -OSD

-t sec Duration of the test in seconds, range: 5 to 86400 (one day, 60*60*24), default 5.
-t60 for one minute, -t3600 for one hour.

-w sec Does x seconds IO without taking these into calculation, to fill up cache, range 2 to 3600, default 0.
Parameter will be ignored by -S1 and –C. –w is part of –t
-t30 -w60: total test duration is 90 seconds, the first 60 seconds will not be taken into account.

-i sec If set, the length of the time-interval between statistic prints, 2-3600, default just one summary after -t sec.
-t300: Prints one result after time has elapsed.
-t300 -i10: Prints IO statistic every 10 seconds, until time of 5 minutes have elapsed.
-t300 -w600 –i10: Total duration is 15 minutes, thereof 10 min wait time, then the first statistics is printed, every 10 sec for the next 5 minutes.

-z % Snooze percent - time spent *sleeping*, default 0. After each completed IO operation the next IO operation starts right after. Snooze delays the next operations with %.
The percentage bases on the average IO response time.

-z 80: Set snooze time to 80%: If the average (read respectively write) IO response time is 5 milli seconds, then the process will wait 4 milli seconds (5 ms * 80%) before starting the next IO. This option will be ignored if a-synchronously IO (-A) is specified.
-z80, -z200

-r % specifies the read percentage, from 0 to 100
0: write only, **100**: read only, **50**: equal read and write, **80**: typical ratio for OLTP.
-r 0, -r50, -r80, -r100

-b size Block size, can be used with K or M, default 4KB, range: 512 Byte ... 64 MB.
-b 512, -b1024, -b1k, -b32k, -b1M

-A num Do a-synchronous IO (pthreads library), range 1-128 concurrent IOs per process (-M flag), default no AIO.
-A4, -A8
-M8 -A10 : creates 8 IO threads, with 10 AIO each → 80 concurrent IOs.

-s size file size, only needed for raw devices, use with K, M or G
-s 256M, -s4G

-c Print IO statistics from /proc/diskstats (Linux only).

-q Be quiet, no additional info is displayed.

-Q Suppressing of header line, and sets -q.

-P Prints just the header line and exits. Optional -x or -c must be specified before -H.
Useful for batch jobs, if output is piped into files.

-x srw Print distribution of IO response times in 1000th (%%). ___ indicates zero IO in this range,
--- indicates less than 1%% (0.1%)
arguments: r=read (only) w=write (only) s=read and write together. If read is set to 0% (-r0, write only) then w will be set, if read is set to 100% (-r100, read only) then r will be used.
-xr, -xw, -xr

-V Summary output, sets -q, also prints 0 instead of ___ or --- for spreadsheets.

-T "text" Any text you want to display at the end, e.g. for batch jobs. The quotation marks " " are mandatory!
-T "Run 1 out of 5"

-l LANG Set the LC_NUMERIC value to LANG, e.g. to print integer 1234 as 1,234 (en_US) or 1.234 (de_DE), or floating point 3222111.44 as 3,222,111.44 (en_US) or 3.222.111,44 (de_DE).
-l en_US, -l de_DE

-L log Writes the output to log file in addition, file will be created and appended.

-d Provides some workflow information

Distributed workload generation

xdisk has the capability to start itself via ssh on remote hosts, to do so ssh-keys needs to be setup without passphrase. The results from all hosts will be summarized. The used files and directory structure must be the same on all hosts, as specified with -f or -F.
The host xdisk is started from does not do IO. If this host should do IOs, then it must be listed as parameter.
e.g.: Controlling host is DB_server_0 ➔

-n host Starts xdisk on remote >host<
-n DB_server_1

-n list Use "list" of hosts, separated by , or : or #, max number of hosts16.
-n DB_server_1:DB_server_2:DB_server_3
-n DB_server_0:DB_server_1:DB_server_2, DB_server_3 ➔ starts IO processes on controlling hosts as well.

-N file >file< contains list of hosts, one per line. No blank or empty lines.
-N list.txt :
DB_server1
DB_server2
DB_server3

If there is a problem or error during the remote operation, then visit the log and err files on the appropriate hosts: /tmp/xdisk.*data-time*.out, /tmp/xdisk.*date-time*.err

Examples

- **OLTP workload**
  `-R -r 80 -b 16k, 32k -M 4 ... 32`
- **Video broadcasting,** streaming read
  `-S1 -r 100 -b 1M -M 1 ... 8`
- **Video creation,** streaming write
  `-S1 -r 0 -b 1M -M 1 ... 8`
- **SAP™ HANA™,** startup, random read
  `-R1 -r 100 -b 256k, 1M -M 4 ... 8`
- **SAP HANA,** normal operation, random write
  DATA: `-R -r 0 -b 256k, 1M -M 4 ... 8 -t300`
  LOG: `-S -r 0 -b 16k, 256k, -M 1 ... 4 -t300`
- **Backup Server,** random read/write
  `-R -r 50 -b 16k, 64k, 256k -M 4 ... 8 -t300`

The file size for all examples should be in the range of approx. 1GB for high -M number up to 20GB for low -M number.

For -R or -S use a meaningful duration like 300 to 600 seconds, not the default of 5

Use -OD always, or -ORW on AIX

If you have only one mount point to measure you can use: **`-f /mount_point/F...`**
if you have multiple mount points you need to specify them separately, typically for Db2 or Oracle:
`-f /db2/data0/F1,/db2/data0/F2,/db2/data0/F3,/db2/data1/F0,/db2/data1/F1,....`

Better would be to put all files into one file and use the parameter **`-F file_list.txt.`**

Creation of workload from 4 hosts: host1, host2, host3, host4, in total 16 IO processes

```
xdisk -n host1,host2,host3,host4 -R -r50 -M4 -f/data/F... -b64k -OD -V -Q -t300
```

Performance evaluation:

Run your type of workload, and increase the number of IO threads until the maximum IO rate has been reached:

```
xdisk –R -r80 -b 16k -M x -f /sap/oracle/data/F... -w600 -t 300 -V -OD -L /tmp/xdisk.txt
```

Here, OLTP, start with -M8, and increase the number like, 16, 32, 64

For write only or read only operations:

```
xdisk –R -r0   -b 256k -M x -f /sap/oracle/data/F... -w600 -t 300 -V -OD -L /tmp/xdisk.txt
xdisk –S -r100 -b 256k -M x -f /sap/oracle/data/F... -w600 -t 300 -V -OD -L /tmp/xdisk.txt
```

start with -M1, and increase the number like 2, 4, 8, 12, 16, ...

Output

These 16 columns are always displayed:

BS	Proc	AIO	read%	IO	Flag	IO/s	MB/s	rMin-ms	rMax-ms	rAvg-ms	WrAvg	wMin-ms	wMax-ms	wAvg-ms	WwAvg
1M	2	0	0	S	D	1590	1590	0.0	0.0	0.0	0.0	1.20	2.54	1.26	1.25

BS: block size in K Byte or M Byte
Proc: number of IO process, as specified with –M, here **2**
AIO: number of async IO, here no async IOs.
read%: read percentage, here **0**% → write only
IO: IO type as specified with -**S** or -R; -C sets S, -C together with -R sets R
Flag: -O flag, here -O**D**: O_**D**IRECT
IO/s: measured IO/s, here **1590** IO per second
MB/s Calculated throughput: IO/s * block size, here **1590** MB/s

The following columns are displaying the IO response time statistics:

```
W  r  Min
|  w  Max
|  |  Avg
|  |   └──► Minimum / Maximum / Average
|  └─────► read or write
└────────► Weighted Average
```

Min: The shortest measured IO response time of all IO
Max: The longest measured IO response time of all IO
Avg: The calculated average (mean) IO response time of all IO
W_Avg: The calculated weighted average (mean) IO response time of all IO, the weighting factor is the IO distribution.

Example: 90 IO with a response time of 5ms, 80 IO with 6ms, 2 IO with 500ms:

```
Avg :     (90 *5    + 80 *6    +  2 *500 ) / (90+80+2) = 11.2 ms
           52%*5ms + 47%*6ms +  1%*500ms
```

Weighting factors = 0.52 0.47 0.01

WAvg = $\frac{0.52^2*5 + 0{,}47^2*6 + 0.01^2*500}{0.52^2 + 0{,}47^2 + 0.01^2}$ = 5,6 ms

The weighted Average indicates the most likely IO response time.
In this example 99% of all response times are in the range between 5 ms and 6ms, the average mean of 11ms does not indicate this. If WAvg is much less than Avg, then there are some IO that takes very long.

If -x is specified these data are displayed in addition:
Distribution of read & write response times:

```
us1   2   4   8  16  32  64 128 256 512 ms1   2   4   8  16  32  64 128 256 512  s1
___ ___ ___ ___ ___ ___ ___ 995   1   1 --- --- --- --- --- ___ ---  ___ ___ ___ ___
___ ___ ___ ___ ___ ___ ___ 867 130 --- --- ___ --- ___ --- ___ ---  ___ ___ ___ ___
```

During this run 99,5% of all reads had a response time between 128 µs and 256 µs (micro)
A few IO had a response time between 256 µs and 1 ms (milli).
A very few IO (less than 0.1%) had a response time between 1ms and 126 ms (milli). During this run 87% of all writes had a response time between 128 µs and 256 µs, 13% were between 256 µs and 512 µs.

If -V is specified in addition to -x, then the write distributing is not displayed in a second line, but after the read distribution.

```
us1  2  4  8 16 32 64 128 256 512 ms1  2  4  8 16 32 64 128 256 512  s1 us1  2  4  8 16 32 64 128 256 512 ms1  2  4  8 16 32 64 128 256 512  s1
  0  0  0  0  0  0  0 984   1   3  4  0  0  3  0  0  0  0  0  0  0    0  0  0  0  0  0  0 717 281   0  0  0  0  0  0  0  0  0  0  0  0
```

Sample screen output:

```
xdisk -C10m -i5 -b64k -OD -f /backup/F...
        IO Disk test
        No. IO threads    : 1
        No. Async threads : 0
        I/O type          : Create file
        I/O type          : Sequential
        Block size        : 64 KB
                          : Write Only
        Start time        : 2016.09.08-08:13:21
        Sync type         : no fsync().
        File open Flag    : O_DIRECT
        Number of files   : 1
        File size         : 10 MB, 0 GB
        Run time          : 0 seconds
        Snooze %          : 0 percent
        File Names        : "/backup/F..."
    BS Proc AIO read% IO  Flag    IO/s   MB/s rMin-ms rMax-ms rAvg-ms   WrAvg wMin-ms wMax-ms wAvg-ms   WwAvg
   64K    1   0     0 SC    -D    2238  139.9     0.0     0.0     0.0     0.0   0.361   0.736   0.414   0.410
```

```
xdisk -R0 -r50 -t30 -i5 -b4k -OD -f /backup/F...
        IO Disk test
        No. IO threads    : 1
        No. Async threads : 0
        I/O type          : Random
        Block size        : 4 KB
                          : Equal read and write
        Start time        : 2016.09.08-08:13:58
        Sync type         : no fsync().
        File open Flag    : O_DIRECT
        Number of files   : 1
        Run time          : 30 seconds
        Snooze %          : 0 percent
        File Names        : "/backup/F..."
    BS Proc AIO read% IO  Flag    IO/s   MB/s rMin-ms rMax-ms rAvg-ms   WrAvg wMin-ms wMax-ms wAvg-ms   WwAvg
    4K    1   0    50 R     -D    4388   17.1   0.141    10.6   0.182   0.177   0.203    14.1   0.267   0.256
    4K    1   0    50 R     -D    4427   17.3   0.143    7.38   0.181   0.177   0.200    6.83   0.264   0.255
    4K    1   0    50 R     -D    4371   17.1   0.142    1.58   0.182   0.178   0.205    6.72   0.269   0.256
    4K    1   0    50 R     -D    4368   17.1   0.138    7.50   0.182   0.178   0.208    21.0   0.269   0.255
    4K    1   0    50 R     -D    4302   16.8   0.139    6.90   0.182   0.178   0.201    24.0   0.276   0.256
    4K    1   0    50 R     -D    4061   15.9   0.140    8.66   0.180   0.177   0.207    37.5   0.305   0.255


xdisk -R -r50 -b8k -f /data/F1 -t10 -l de_DE -V
    BS Proc AIO read% IO  Flag    IO/s   MB/s rMin-ms rMax-ms rAvg-ms   WrAvg wMin-ms wMax-ms wAvg-ms   WwAvg
    8K    1   0    50 R      - 521.258  4.072   0,001   0,106   0,001   0,005   0,001   0,148   0,001   0,005
```

Disclaimers

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THE xdisk TOOL AND THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.