OpenSFS
Open Scalable File Systems, Inc.

OpenSFS Benchmarking Work Group (BWG)

Scalable parallel file systems I/O workload characterization survey

The OpenSFS Benchmarking Work Group (BWG) (http://www.opensfs.org/) aims to provide an I/O benchmark suite to satisfy the benchmarking requirements of the scalable parallel file system users and facilities. Towards this end, the workgroup aims to characterize the I/O workloads from small- to very large-scale scalable parallel file systems deployed at various high-performance and parallel computing facilities and institutions. Using these gathered characteristics, the workgroup will identify and build the required I/O benchmarks to emulate these workloads and provide required documentation about the benchmark suite. Your assistance is needed to provide a characterization of the I/O workload at your facility by completing the survey below. *Collected data shall not be used for marketing or sales purposes*. Gathered raw data based on the answers provided to the BWG survey will only be available to the OpenSFS BWG members. The anonymized data based on the answers gathered from the BWG survey will be available to the public at the discretion of the participants. While there are no mandatory fields, the BWG appreciates any and all answers. Please contact bwg-survey-data@opensfs.org for questions or comments or completed survey data.

Target release date for the community: August 13, 2012
Deadline to gather the answers from the community: September 31, 2012

On behalf of the OpenSFS BWG community,

Richard Vanderbilt and Sarp Oral
BWG Co-chairs

1. Site Information

- Site name:
- Affiliation or funding agency/division:
- Site contact:

  Telephone number:
  Email address:
- How many different HPC systems/platforms operated concurrently?
- Number of overall users:
- Number of users of file systems(s):
- Please select all that apply as predominant site activities/functions for your facility:

  Bio-science research and industry
  National research laboratory
  National security
  Academic (not for profit)
  Energy
  Media/entertainment
  Other (please explain)
- Are there any restrictions on releasing your data to the public after its been anonymized (site and application identifiers removed)?

2. File System Configuration
(*duplicate as needed for multiple scalable parallel file systems*)

Architecture

- Scalable parallel file system being used?
  Version?
  How long the file system has been in production?
- Architecture of the scalable parallel file system:
  Acquired as a turn-key file system (e.g. a file system appliance)?
  Acquired as parts and built in-house?
- Number of clients using the scalable parallel file system?
- Number of CPU cores using the file system?
- Is there an I/O forwarding mechanism for compute cores to access the file system?
- Are file system clients based on virtual machines (VM), if so how many VMs per actual, physical hosts?
- Connectivity diagram of file system, servers and clients (please attach to document if available):

Hardware

- Disk types and interfaces (FC, SAS, SATA, NVRAM)?
- Redundancy configuration of disk groups in file system (if applicable)?
- Interconnect between I/O controller(s) (e.g. RAID) and I/O servers (e.g. FibreChannel, SAS, Infiniband)?
- Interconnect topology (e.g. fat tree) and type (e.g. Infiniband) between I/O servers and scalable parallel file system clients?
- Interconnect topology (e.g. torus, fat tree) and type (e.g. Infiniband, Ethernet, Gemini) in the compute platform(s)?
- The age of the system (disks, controllers, servers, clients)?
- Are there any other relevant hardware details that can be shared?

Usage

- Size of file system (in TB - raw and available) (SI)?
- Number of total files and directories on disk?
- Large vs. small files:
  Number of files with size < 64 KB?
  Number of files with size >= 64 KB and < 1 MB?
  Number of files with size >= 1 MB and < 1 GB?
  Number of files with size >= 1 GB and < 100 GB?
  Number of files with size >= 100 GB?

Please note that the above statistics (and far more) can be retrieved automatically for the whole file system through the use of the fsstats script available from the Petascale Data Storage Institute (http://www.pdsi-scidac.org/fsstats/). This can be run on any client with access to the file system of interest, including directly on a mounted client on the metadata server (e.g. MDS for a Lustre file system) for best performance. Note however, that the script currently runs at *the maximum rate possible, which can impose a significant meta data load and may impact other users of the filesystem.*

- Characteristics of file system (e.g. is it primarily used for checkpointing)?
- Is the file system shared among multiple compute platforms and if so by how many:
- How many applications are typically performing I/O concurrently per file system?
- What is the aggregate total amount of I/O issued to the file system for a day?
- Do the application workloads change during the day or night or over the weekends?
- Is file system level I/O typically composed of small or large I/O blocks?
- What is the file system read/write request ratio?
- Is there heavy metadata usage for the typical application mix? Why (How do you determine this)?
- Are the applications generating predominantly sequential or random I/O patterns (observed aggregate at the file system level)?
- Does your facility gather usage and performance statistics for the file system (see fsstat above and Appendix A, for example)?

  If so, can these statistics be included as an attachment to this survey?
  What tools were used for gathering these statistics?

3. Application I/O profile
(*duplicate as needed per application*)

- Application name?
- Application domain?
- Application I/O characteristics:

    I/O type (e.g used for application checkpointing or as a part of normal I/O for computation)?
    I/O request sizes (if known)?
    I/O request pattern (e.g. random or sequential)?
    File access pattern (e.g file per process N:N, shared file N:1, or mixed M:N)?
    Number of threads doing concurrent I/O or metadata operations?
    Number I/O operations per application runtime (excluding meta data operations)?
    Number of metadata operations per application runtime?
    Typical runtime of the application?
    Application I/O libraries used (e.g POSIX, MPI-IO, or other high-level libraries HDF5, ADIOS, pNetCDF)?
- Any other application I/O characteristics needs mentioning?

Appendix A: Scripts and recommendations

Lustre/Linux configuration:

A Lustre/Linux specific sample data collection script, to be run on the MDS, one OSS, and one representative client. Running this script once collects some of the above requested info and some additional Lustre tunings. It can be run as any regular user, and does not require root privileges. The script generates an output file named “opensfs-survey.text” in the /tmp directory. Please inspect the file contents upon completion of the script execution and send back the results to OpenSFS BWG as a part of this survey.

```
# general information about the system
{
date
uname -r
tail -25 /proc/cpuinfo
grep Mem /proc/meminfo
cat /proc/mdstat
# information about mounted filesystems
mount -t lustre
df -h -P -t lustre
df -h -P -i -t lustre
# Lustre node configuration
lctl get_param -n nis
lctl get_param -n version
lctl get_param -n devices | grep -c OST
lctl get_param *.*.num_exports
lctl get_param -n osc.*.checksum_type | sort -u
lctl get_param -n osc.*.max_rpcs_in_flight | sort -u
# IO/RPC statistics from running server threads
lctl get_param -n *.*.*.stats | egrep -h -v "snap|req|llog" | sort
lctl get_param -n *.*.brw_stats | egrep -B1 -A9 -m2 "bulk|I/O size"
} 2>&1 | tee /tmp/opensfs-survey.txt # log output to file
```

Also, for Lustre deployments the Lustre monitoring tool (LMT) can be used to gather online monitoring data about the scalable parallel file system statistics and usage. The Lustre Monitoring Tool (LMT) is a collection of plug-in modules for the Cerebro data transport utility. Cerebro is a light-weight daemon that runs on each Lustre server node and manages a set of data collectors. Each data collector is a C-based library, and the LMT data collectors gather information from /proc every five second, package it, and forward it to a MySQL-based database. From there the data can be presented in near real-time, or mined for past performance and behavior. More details on LMT can be obtained from https://github.com/chaos/lmt/wiki or https://github.com/chaos/cerebro/wiki.

Another useful tool to collect statistical and usage data for a Lustre file system deployment is the collectl. More information on collect can be found at http://collectl.sourceforge.net/ or http://collectl.sourceforge.net/Tutorial-Lustre.html or http://collectl.sourceforge.net/Lustre.html.

GPFS configuration:

For GPFS scalable parallel file systems mmpmon facility can be used to gather some of the data asked in this survey. Below is the overview of mmpmon from the "Overview of mmpmon" document in the GPFS Advanced Administrators Guide:

The mmpmon facility allows the system administrator to collect I/O statistics from the point of view of GPFS servicing application I/O requests.

The collected data can be used for many purposes, including:

- Tracking I/O demand over longer periods of time - weeks or months.
- Recording I/O patterns over time (when peak usage occurs, and so forth).
- Determining if some nodes service more application demand than others.
- Monitoring the I/O patterns of a single application which is spread across multiple nodes.
- Recording application I/O request service times.

For more details on using mmpmon facility please refer to:
https://publib.boulder.ibm.com/infocenter/clresctr/vxrx/topic/com.ibm.cluster.gpfs.v3r5.gpfs200.doc/bl1adv_mpmover.htm