U.S. Department of Energy Best Practices Workshop on
File Systems & Archives
San Francisco, CA
September 26-27, 2011
Position Paper

Stephan Graf
Jülich Supercomputing Centre
st.graf@fz-juelich.de

Lothar Wollschläger
Jülich Supercomputing Centre
l.wollschlaeger@fz-juelich.de

ABSTRACT / SUMMARY

The storage configuration for the supercomputer *JUGENE* in Jülich consists of a GPFS cluster (*JUST*) and two Oracle STK SL8500 tape libraries. In this paper the actual configuration and the next upgrades are described. Furthermore a project for using flash storage as a kind of cache memory for the disk storage is introduced.

INTRODUCTION

The Jülich Supercomputing Centre (JSC) operates two supercomputer: The BlueGene/P System *JUGENE* and the x86 based *JuRoPA* system. While the *JUGENE* uses the remote GPFS cluster *JUST*, the *JuRoPA* users works on a local Lustre based storage. There the users can access their files in the GPFS file system via dedicated nodes.

The consideration in this paper for the actual and the future storage configuration/implementation are focused on the *JUGENE* and the *JUST* GPFS cluster.

The users can access three types of file systems:

On $HOME they should store there code and develop their program.

For the job run they are urged to use the scratch file system $WORK to get the maximum IO performance.

To archive their results the data should be moved to the $ARCHIVE file system.

JUGENE STORAGE PERFORMANCE TODAY

The *JUGENE* is build up of 72 BlueGene/P Racks with 1 PF peak performance and 144 TiB main memory. Each rack contains 1024 compute nodes (CN) and 8 IO nodes (576 IO nodes in total), with each one connected via 10GbE to the *JUST* storage. Measurements show that a single IO node gets an IO performance of 450 MB/s reading and 350 MB/s writing. For a whole rack it is 3.6 GB/s reading and 2.8 GB/s writing. The maximal peak IO for the full system is 260 GB/s reading and 200 GB writing. Assuming that 50% of the main memory of one rack (1024 CN) is to be written on file system (e.g. for checkpointing), the required time is 5 minutes for reading and 7 minutes for writing. To write 50% of the main memory of the full system in 15 Minutes requires 0.5 *144 TiB /1800s = 44 GB/s. The *JUST* cluster based on DS5300 storage devices provides 66 GB/s. But only half of the cluster is used for the fast scratch file system $WORK. The other half of the clusters hosts the $HOME and $ARCHIVE file system. This implicates that the $WORK can be saturated by 33 GB/s / (8*0.35 GB/s) = 12 racks writing to the file system.

On the *JUST* cluster 8 building blocks provides the $WORK file system, each containing a DS5300 with 36 LUNs per DS5300 having a size of 8 TB (RAID6). This leads in a total capacity for $WORK of 2.3 PB.

BLUEGENE/Q INSTALLATION IN 2012

In 2012 the *JUGENE* will be replaced by a BlueGene/Q system consisting of 6 racks. There are 8 IO nodes per rack, each having a dual 10GbE port with an aggregated bandwidth of 1.5

GB/s. So the maximum throughput of a rack is 12 GB/s and the full system 72 GB/s.

If 50% of the main memory of on rack (1024 CN with 16 GB RAM per node) are to be written on disk it will last (approximately) 12 minutes. So to write 50% of the full system main memory to the storage in 15 minutes, a bandwidth of 50%*384 TiB /1800s = 115 GB/s are required. Therefore we will get a storage upgrade for the *JUST* GPFS cluster. We are planning to install 8 DDN SFA12000 and getting an aggregated bandwidth between 100GB/s and 160 GB/s for the scratch file system $WORK. The performance for $HOME and $ARCHIVE will also increase, but this is not concerning us.

FLASH MEMORY AS SCRATCH FILE SYSTEM

In parallel the JSC will investigate a new storage concept using flash memory cards as a kind of cache between the IO nodes and the ordinary disk storage. It is a European Union funded project, a PRACE (Partnership for Advanced Computing in Europe) prototype for next generation supercomputers.

4 x86 systems each with 2 fusionIO ioDrive Duo 320GB SLC will be set up. The bandwidth of the flash card is 1.5GB/s. The cumulated performance of these 4 nodes should be 12 GB/s, a similar value as 8 BlueGene/Q IO nodes (one rack). Using the GPFS features to setup different kind of storage pools and to implement placement and migration policy rules a concept will be modeled, that new created files will be created on flash, and GPFS will migrate the files to disk in the background automatically.

This concept will be implemented with real BlueGene/Q hardware as soon as it is available.

ARCHIVE STORAGE EXPANSION

The users should store their results on the $ARCHIVE file system. There the data will be migrated by Tivoli HSM on tapes (weighted by *size* and *last access time*). For safety two versions (COPYPOOL) will be held on tape. Furthermore every file of the $HOME and the $ARCHIVE file system will be backed up. Files on the scratch file system $WORK are not backed up and will be deleted after 90 days.

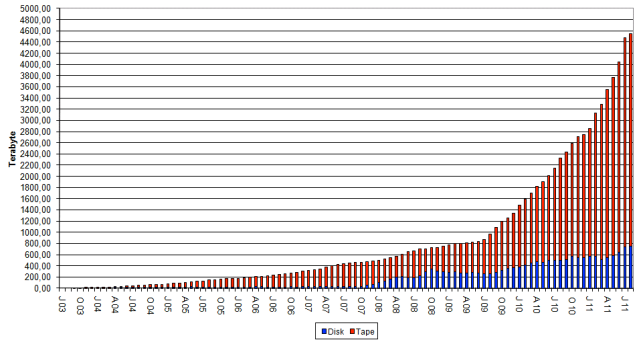

Figure 1: Data groth on the GPFS cluster JUST

The JSC operates two Oracle STK SL8500 libraries with an aggregated capacity of 16.6 TB (T10K-B tape drives). In figure 1 the exponential data growth on our storage cluster can be seen. We expected to run out of space in the third quarter 2011. Because of the ordering and shipping delay of the new hardware it became critical the last month. But now the new hardware has arrived and is going in production. 16 T10K-C tape drives have been added and the new tape generation (which is able to store 5 TB) will replace the old tapes step by step.

This kind of upgrade is the typically way for us to manage the growth of data amount. For the next 6 years we plan to enlarge the capacity of the two libraries to 80 PB just by upgrading to the next tape drive generation T10K-D.

CONCLUSIONS

On our supercomputer a specific maximal I/O performance is available and for the user it is reasonable to get the maximum performance from the file system. But this is often difficult to achieve. Therefore it is mandatory to train the users and give them the knowledge to speed up their jobs I/O. For this purpose we have developed the *SIONlib* in Jülich. The users can use this library in there code to map very easily local task I/O to one file. By using the *SIONlib* it is possible to get nearly 100% of the performance on the scratch file system $WORK from the JUGENE. We also use this tool for benchmarking parallel file systems.[1]

The other subject concerns the long time data storing. Till now and for the next years we are able to store all user data in our archive system. The new technologies keep up with the data growth in Jülich. But there are upcoming questions like how long must the data be hold or what happens when a project ends. These problems must be tackled in mid or long term.

REFERENCES

. [1] http://www.fz-juelich.de/jsc/sionlib/