Methodology to select a I/O configuration (hardware resources and stack software) in cloud platform

Pilar Gomez-Sanchez, Dolores Rexachs
Computer Architecture and Operating Systems Department (CAOS)
Universitat Autònoma de Barcelona, Bellaterra (Barcelona), Spain.
{pilar.gomez, dolores.rexachs}@uab.cat

Abstract- *Currently, there is an increasing interest about the cloud platform by the High Performance Computing (HPC) community, and the Parallel Input/Output (I/O) for High Performance Systems is not an exception. However, in cloud platform increase the number of parameters that user can select to I/O system. For this reason, we propose a methodology to help the user to select a configuration.*

I. INTRODUCTION

Nowadays, the interest about the cloud computing platform is increasing. The scientific community have interest about the cloud computing because some benefits of clouds are that users can acquire and release resources on-demand and they can configure and customize their own Virtual Cluster (VC) [1]. Parallel scientific applications that use parallel I/O can benefit of these platforms, because the user can create and configure the I/O system considering the application requirements which represents an advantage over the traditional HPC-IO systems. But, in cloud platform increase the number of parameters that user can select to I/O system.

A. Methodology

For this, we propose a methodology to configure and select a configuration depending application requirements and user requirements. Our methodology has six steps:

1) Application Parallel I/O Characterization: The I/O characteristics are represented by an I/O model. We trace the parallel application with PAS2P-IO [2] and the traces are analyzed to obtain the I/O model. The I/O model allows us to meet the minimum application requirements.

2) Creation and Configuration of the Virtual Clusters: A VC is represented by the components shown in Table I. We can create a VC quickly with StarCluster [3]. We apply the following considerations as a starting point on the selection of the components for a VC that meets the user requirements.

3) Characterization of the Virtual Clusters: We use the IOzone [4] benchmark to obtain the average values for the transfer rate at local file system level. Normally, the user can check if he will obtain the waited performance.

4) Performance Evaluation on the Virtual Clusters for the application I/O model: IOR [5] benchmark evaluates the performance at global file system level. IOR is designed to measure parallel file system I/O performance at both the POSIX and MPI-IO level. The IOR performs writes and reads to/from files under several sets of conditions and reports the resulting throughput rates.

We analyze the access patterns of the I/O model at phases

Parameters	Description
Instance Type(*)	Number of cores, processor capacity, RAM memory size
Number of instances (*)	
Number of I/O nodes (-)	Data servers and metadata server.
Storage type(+)	Temporal and/or persistent.
Device type temporal (+)	HDD or SSD.
Device type persistent (+)	HDD or SSD.
Capacity of temporal storage(+)	As minimum the storage capacity required (expression 4).
Capacity of persistent storage(-)	
Network performance (+)	Low, Moderate, High, Unknown.
I/O Library (-)	MPI, NetCDF, pnetcdf, HDF5.
Local file system (+)	File system Linux ext3, ext4, xfs, etc.
Global file system (-)	Parallel, Distributed or Network file systems.
Stripe size (-)	Related by the parallel file system.

(*) the parameters which can be selected by the user, (-) the parameters that the user must configure manually, (+) the parameters that the user cannot change because they are by default depending on instance type.

level and proposed an IOR configuration based on the application I/O model, where the relevant parameters are the numbers of processes (np), the number of segments (-s), block size (-b) and transfer size (-t). Table II shows input parameters for IOR based on the I/O model phase. The output of this process is the transfer rate expressed in MB/ sec, named BW_{CH}, and I/O time for application I/O model. The I/O model has been extracted executing the application once in the cluster.

5)Cost Evaluation of the Virtual Clusters: Performance obtained using IOR for the application I/O model is used to calculate the cost. The total cost for a specific VC is composed of a variable cost and a fixed cost.

6) Comparison of the Performance-Cost Ratio for the Virtual Clusters: The performance and the cost for the VCs are presented to the user to simplify the decision making. To compare the performance-cost of the different VCs.

II. EXPERIMENTAL RESULTS

In this section, we present the performance evaluation and the cost analysis for two scientific application such as I/O kernels NAS BTIO and S3D-IO that present different I/O access patterns. BT-IO and S3DIO have been traced using PAS2P-IO to extract their I/O models. The I/O model has been extracted executing the application once in the cluster.

1) Application Parallel I/O Characterization: The Figure 1 shows the BT I/O model and the Figure 2 shows the S3D-IO I/O model.

NAS BT-IO - 16 Processes - FULL Subtype - Class B - A SHARED File

Zoom of the READ phase
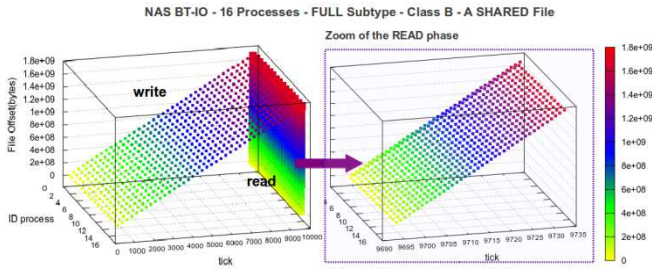
Fig. 1. The left picture shows the I/O model for the application and the right picture shows a zoom on the read operations. It can be observed that write and read are done in the same file offset. The application uses a shared file. Each MPI process performs a write operation every 122 communication events. This is done 40 times, and after, each process performs 40 read operations consecutively.

S3D-IO – 200 x 200 x 200 – Five SHARED Files

8 Processes (2 x 2 x 2)   16 Processes (2 x 2 x 4)
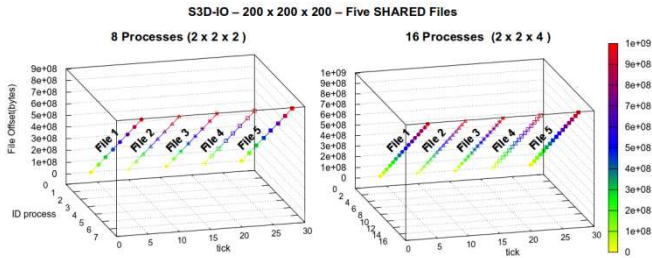
Fig. 2. The left picture shows the I/O model for 8 processes with a workload 200x200x200. The application uses five shared Files. All MPI processes write once on the File 1, after all processes write on the File2, and so on. This access pattern is representing the checkpointing process for the S3D application. The same behavior is observed in the right picture for 16 processes.

2) Creation and Configuration of the Virtual Clusters: We select some instances depending application and user requirements. See Table II and Table III.

Table II. Characteristics of the Amazon's instances selected

Instances	Processor	CPU	RAM (GB)	Storage(GB)	AWS Ireland ($ Per Hour)	AWS Virginia ($ Per Hour)
m1.small	Intel Xeon Family	1	1.7	1x160	0.047	0.044
m1.large	Intel Xeon Family	2	7.5	2x420 HHD	0.190	0.175
c3.xlarge	Intel Xeon E5-2680 v2 2.8 GHz	4	7.5	2x40 SSD	0.239	0.210

3) Characterization of the Virtual Clusters: We use this phase to meet the instance performance.

4) Performance Evaluation on the Virtual Clusters for the application I/O model: For example, Table IV show the input parameters to configure IOR from the I/O model phases of the BT-IO. From this process, we obtain the transfer rate (BW_{CH}) and execution time for the BT-IO model. These values are used to calculate the variable cost.

5) Cost Evaluation of the Virtual Clusters: in this step we calculate the cost to use a determinate I/O configuration.

6) Comparison of the Performance-Cost Ratio for the Virtual Clusters: Finally, we compare what configuration is better depending of Performance-Cost ratio. We show the obtained result to BT in Figure 3

III. CONCLUSIONS

In this paper, we have proposed a methodology to help the user to configure and select the configurations in a cloud platform depending of application requirements and the user requirements. Our methodology proposes to customize the IOR benchmark with the I/O model to evaluate quickly different configurations and reducing the evaluation cost because it is not necessary to execute the application every time. As future work, we will continue analyzing the influence
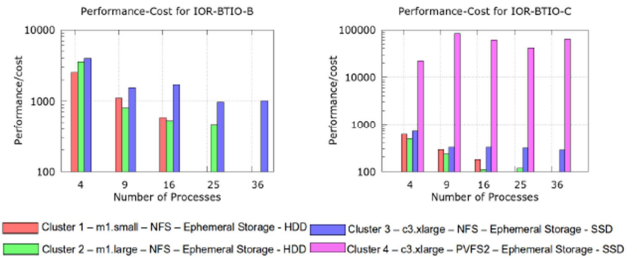
of the different components for the Virtual Cluster configuration.

Table III. Descriptive characteristics of the virtual clusters configured for the experiments (Step2)

I/O components	Virtual Cluster 1	Virtual Cluster 2	Virtual Cluster 3	Virtual Cluster 4
Instance Type	m1.small	m1.large	c3.xlarge	c3.xlarge
Number of Instances	17	17	11	11
Storage Type Temporal	Ephemeral	Ephemeral	Ephemeral	Ephemeral
Storage Type Persistent	EBS	EBS	EBS	EBS
Device Type Temporal	HDD	HDD	SSD	SSD
Device Type Persistent	HDD	HDD	HDD	HDD
Capacity of Temporal Storage	160GB	420GB	40GB	300GB
Capacity of Persistent Storage	8GB	8GB	8GB	16GB
Networking Performance	Low	Moderate	High	High
Number of data servers	1	1	1	8
Number of Metadata Server	1	1	1	1
File system Local	ext3	ext3	ext3	ext3
File system Global	NFS	NFS	NFS	PVFS2
Stripe Size	—	—	—	64KB
I/O library	mpich2, pnetcdf	mpich2, pnetcdf	mpich2, pnetcdf	mpich2, pnetcdf
EBS Fixed Cost EU($ per GB-month)	0.55	0.55	0.55	0.55
EBS Fixed Cost US-East($ per GB-month)	0.50	0.50	0.50	0.50

Table IV. IOR Input parameters from the I/O model phases of the NAS BT-IO subtype FULL-Collective Operations and s=rep=40. Outputs for the virtual Cluster 3.

np(IdPh)	b=rs(IdPh) (MB)	t=rs(IdPh) (MB)	BW_{CH} (MB/s)	Time (sec)
Class B				
4	10.0	10.0	104	15.5
9	4.5	4.5	91	17.7
16	2.5	2.5	96	16.9
25	1.6	1.6	73	22.6
36	1.1	1.1	74	22.2
Class C				
4	40.6	40.6	87	74.5
9	18.0	18.0	83	77.8
16	10.1	10.1	84	77.2
25	6.5	6.5	82	79.0
36	4.5	4.5	78	82.9

Performance-Cost for IOR-BTIO-B   Performance-Cost for IOR-BTIO-C

Cluster 1 – m1.small – NFS – Ephemeral Storage - HDD   Cluster 3 – c3.xlarge – NFS – Ephemeral Storage - SSD
Cluster 2 – m1.large – NFS – Ephemeral Storage - HDD   Cluster 4 – c3.xlarge – PVFS2 – Ephemeral Storage - SSD

Fig. 3. Performance-Cost ratio of the four Virtual Clusters using IOR configured for the BT-IO. The left picture corresponds to Class B and the right picture to Class C. Results are shown in logarithmic scale. Virtual Clusters with experiments without results are limited by storage capacity or the parallel degree required. Class B was not tested on Virtual Cluster 4 because the I/O workload is small for its I/O system.

REFERENCES

[1] M. Liu, J. Zhai, Y. Zhai, X. Ma, and W. Chen, "One Optimized I/O Configuration Per HPC Application: Leveraging the Configurability of Cloud," in Proceedings of the Second Asia-Pacific Workshop on Systems. ACM, 2011, pp. 15:1–15:5.
[2] S. Méndez, J. Panadero, A. Wong, D. Rexachs, and E. Luque, "A New approach for Analyzing I/O in Parallel Scientific Applications," in CACIC12, 2012, pp. 337–346.
[3] StarCluster. (2014) An Open Source Cluster-Computing Toolkit for Amazon's Elastic Compute Cloud (EC2). http://star.mit.edu/cluster/
[4] W. D. Norcott. (2006) IOzone Filesystem Benchmark. http://www.iozone.org/
[5] W. Loewe, T. McLarty, and C. Morrone. (2012) IOR Benchmark. https://github.com/chaos/ior/blob/master/doc/USER GUIDE