Design and fulfill of system for simulating and visualizing file system

Cong Wang
Institute of High Energy Pyhsics,Chinese Academy of Sciences,Beijing,China
wangcong@ihep.ac.cn

Abstract

Design of a file system simulation and visualization system ,using simgrid API and visualization techniques to help users understanding and improving the file system portion of their application. The core of the simulator is the API provided by simgrid , cluefs track and catch the procedure of the I/O operation. Run the simulator simulating this application to generate the output visualization file, which can visualize the I/O action proportion and time series. Users can change the parameters in the configuration file to change the parameters of the storage system such as reading and writing bandwidth, users can also adjust the storage strategy, test the performance,getting reference to be much easier to optimize the storage system. We have tested all the aspects of the simulator ,the results suggest that the simulator performance can be believable.

Introduction

Simulation has been used for decades in various areas of computing science, such as network protocol design ,microprocessor design. By comparison, current practice in storage simulation is in its infancy. So I am trying to fulfill a simulator with Simgrid to simulate the storage part of application . There are several main concerns about simulating :simulation accuracy ,simulation speed and simulation scalability. We are using the trace caught by cluefs , in the trace file ,each simulated process is described as a sequence of store events. This simulator will simulate these actions with simgrid API and you can check the performance of the storage application by the visualization part of the simulator and you can also find out the suitable configuration parameters for the storage platform by simulating with different configuration files .

Conclusions

Users can use cluefs to get I/O action sequence of the application, and input all the sequence to the simulator, the simulator will simulate I/O action .The simulator has two mode, sleep mode to simulate all the action according to the time in the trace file, it will produce the log including exact executing information and will produce another trace file which is used for visualization ,the visualization part can visualize all the I/O file action sequence of each process to make it clear how the application works on storage part. The second mode is action mode, it simulates all the I/O actions with a platform configuration file .In this file ,users can change the storage structure and storage parameters such as Reading bandwidth of disk. And then users can get the executing time of different platform file to make sure if it is worth to change some configuration of your real storage system. So the simulator is useful when you want to know how the storage part of you application works and when you want to know how your application will work with different disks ,you can simulate it with the simulator and you will gain a lot.

Application of the simulator

LSST is a large optical survey project funded by the National Science Foundation and the Department of Energy. It will continually image the sky, identify changes in near real time, and over a decade of operations collect tens of petabytes of data building up the deepest, widest, image of the Universe. Its data will enable a range of science goals from identification of Near Earth Asteroids to understanding the nature of Dark Energy. A survey of this scale requires significant computing resources but also a modern, high-performance, scalable, data processing and analysis system. It will do some file action which we focus on. So what we want to do is :

1.Simulate the application and visualize to make sure which kind of I/O action it does mostly.

2. Change the parameter of the platform to find out whether it is useful if we put the head of the file in the SSD disk which read much faster, and how much the performance will improve.

We illustrate the declaration of storage components(Fig1) in the XML description format provided by SimGrid depicted in Fig 2.This platform comprises one machine named denise. Two disk are both attached and mounted by this machine.

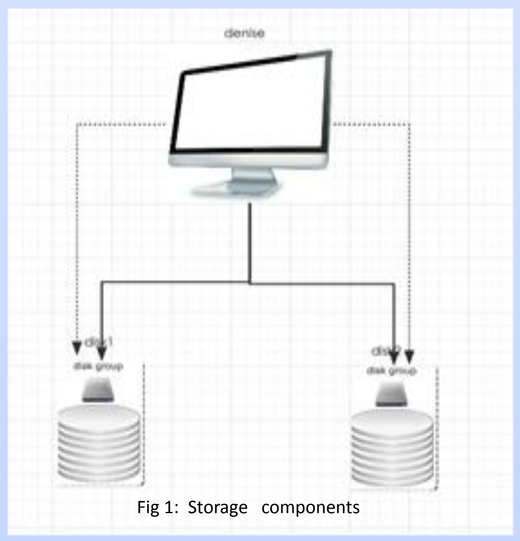


Fig 1: Storage components

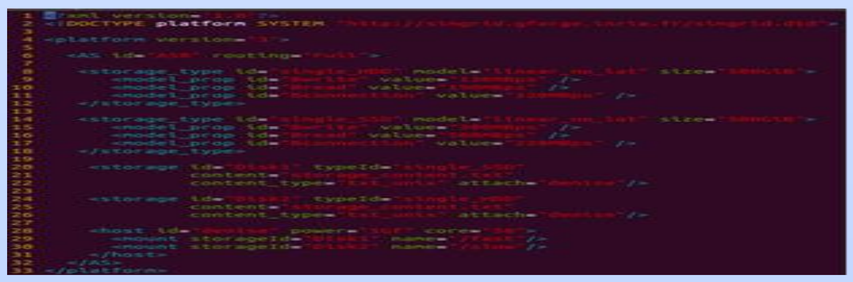
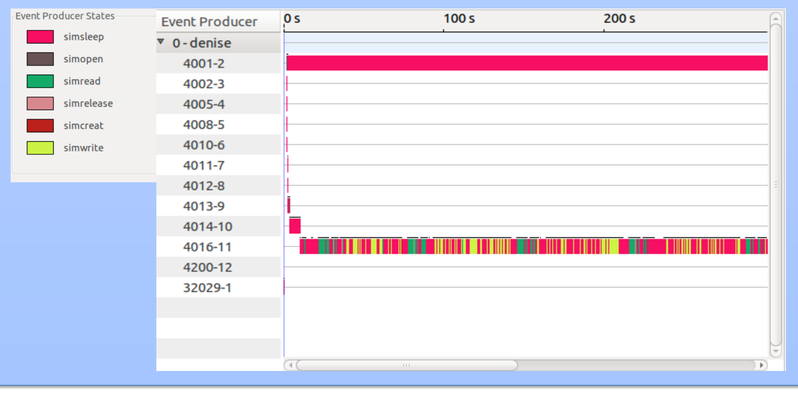


Fig2.storage description in the SimGrid format

Tracing is widely used to observing and understanding the behavior of parallel applications and distributed algorithms. Usually, this is done in two-steps: the user executes the application and the traces are analyzed after the end of the execution. The analysis can highlights unexpected behaviors, bottlenecks and sometimes can be used to correct distributed algorithms.

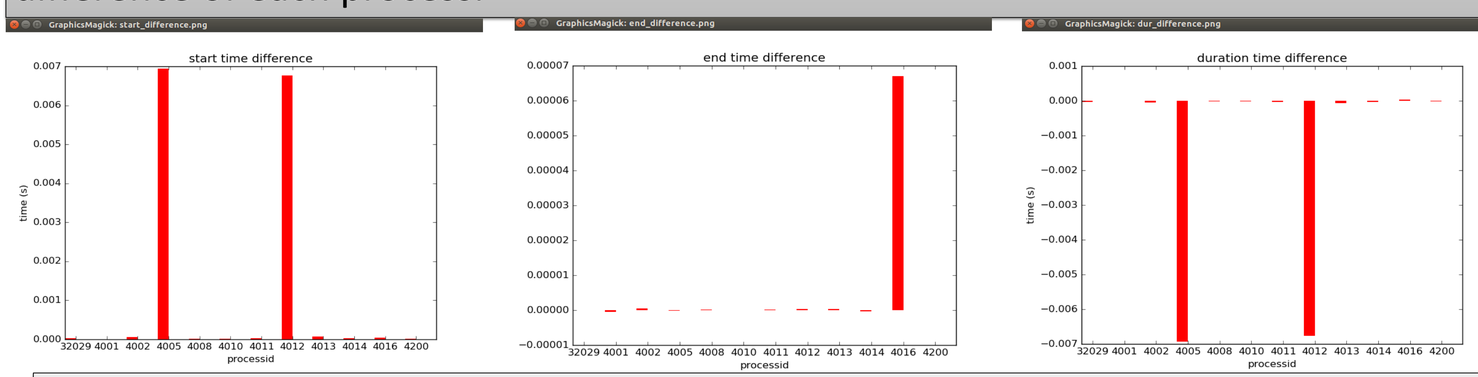
With the trace produced by the simulator ,we can check the action sequence of the application .And we can get the information that which kind of action this application executes mostly directly by the visualization.

Framesoc is a generic trace management and analysis infrastructure. We can use it to visualize the trace file we got after the executing of the simulator and get the picture below.



The accuracy and verification of the simulator

After the executing of simulator ,we can compare the executing time of each process in the simulator with the time in the original trace file, and get the time difference of each process.



The pictures showed above are the time difference between 60_90_300_500（Wbandwidth,Rbandwidth,Wbandwidth,Rbandwidth） platform in sleep mode and the original trace file .It shows that the difference is in millisecond and the whole executing time is more than 600s,so we can receive the difference and we can believe in the result of the simulator.