Enhancing FlashSim Simulations for High Performance Computing Storage Systems

*Omar Rodriguez * (Undergraduate, Junior), Wei Xie and Yong Chen [a] (Advisors)*

[a] Data Intensive Scalable Computing Laboratory, Computer Science Department, Texas Tech University,

**Department of Electrical & Computer Engineering and Computer Science, Polytechnic University of Puerto Rico*

Abstract

- This work describes the design and development of an enhanced FlashSim simulator framework and toolkit.
- The FlashSim is one of the most popular SSD simulators.
- SSD simulator focuses on software components, specifically FTL schemes, garbage collection, and wear-leveling.
- In order to simulate and analyze the performance results, the simulator needs block I/O traces.
- This work extends the FlashSim simulator with leveraging the *blktrace* tool to automate collecting traces from different application workloads.
- We also develop a converter that transforms the I/O traces into the format needed for the FlashSim simulator.
- The proposed extension enhances the FlashSim simulator and automates trace collection, conversion, and analysis.

Introduction

- NAND Flash based Solid State Drives (SSDs) have increasing demands on high-end/high-performance computing systems.
- Unlike the traditional rotating architecture of HDD's, SSD characteristics include:
  - No mechanical components
  - High bandwidth, low latency
  - Shock resistance
  - Low power consumption
  - Superior performance especially for random data accesses
- With the SSDs usage rapidly increasing, most of the design details, however, are not fully revealed to the public domain.
- The research of SSDs have been dominated by using simulators, such as FlashSim.
- To provide stimulus to the simulator and analyze the SSD performances the block I/O traces are needed.
- The *blktrace* is a block layer I/O tracing mechanism which provides detailed information about request queue operations[3].
- In this work, we extend the FlashSim with automating trace collection via using the blktrace tool for application/benchmark workloads including online streaming workload.
- We used the IOzone benchmark to produce different workloads specifically for evaluation.
- The video stream traces were collected for evaluation as well and for studying the impact of I/O traffic in the computers system.
- The collected traces were translated with a converter into the FlashSim format in order to simulate and analyze new SSD design performance.

Design

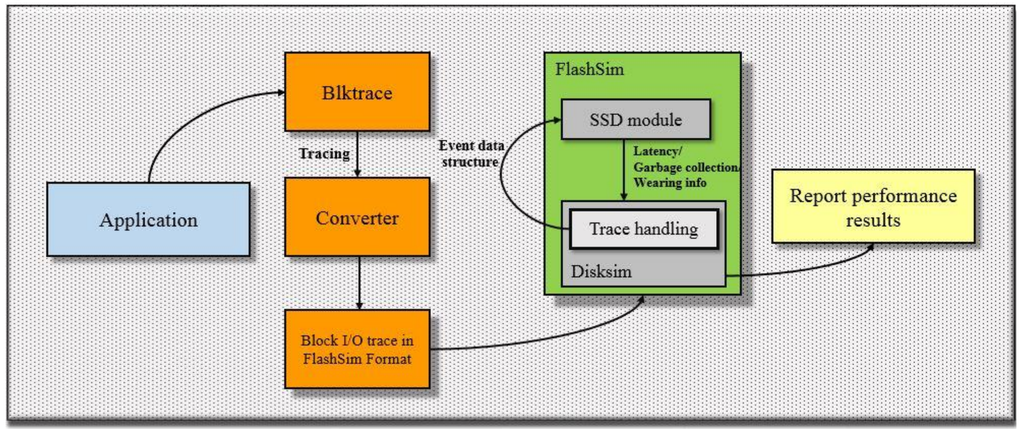

Figure 1: Design Architecture To Enhance The FlashSim Simulator.

Results

```
8,0 0  99 2.050693491 3440 G  R 568896 + 24 [firefox]
8,0 0 102 2.050697151 3440 Q  R 568992 + 16 [firefox]
8,0 0 103 2.050697845 3440 G  R 568992 + 16 [firefox]
8,0 0 105 2.050698545 3440 Q  R 569024 + 8 [firefox]
8,0 0 106 2.050698895 3440 G  R 569024 + 8 [firefox]
8,0 0 107 2.050700021 3440 I  R 568896 + 24 [firefox]
8,0 0 108 2.050700707 3440 I  R 568992 + 16 [firefox]
8,0 0 109 2.050701081 3440 I  R 569024 + 8 [firefox]
```

```
2.050700021 0 568896 24 1
2.050700707 0 568992 16 1
2.050701081 0 569024  8 1
2.932513809 0 9280800 16 1
2.932514455 0 9280840 16 1
2.932514800 0 9280872 8 1
2.932514989 0 9280888 88 1
2.958977465 0 491080 32 1
2.958978218 0 491120 8 1
2.958978678 0 491152 8 1
```

Figure 2: Conversion result from the blktrace format to the FlashSim format.

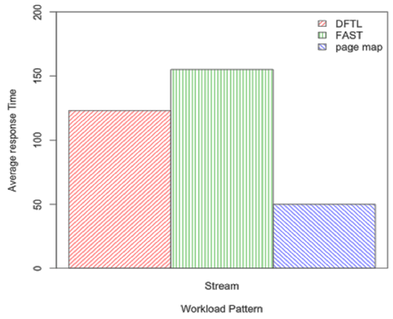

Figure 3: The response time simulation result of Streaming media workload trace.

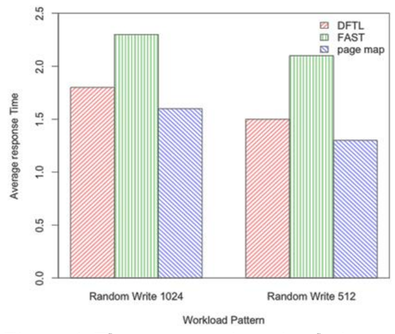

Figure 4: The response time simulation result of IOzone random write workload trace.

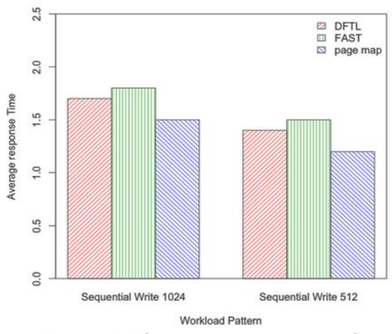

Figure 5: The response time simulation result of IOzone sequential write workload trace.

Experiment

- The experiments were conducted in the Data-Intensive Scalable Computer Laboratory at Texas Tech University.
- We used the IOzone file system benchmark and the online streaming to produce different workloads.
- The video streaming workload was produced using Firefox web browser allowing the stream to run for 10 minutes.
- Simultaneously, the I/O traffic of the workloads were traced using blktrace tool extension for FlashSim.
- The output file obtained from the blktrace was automatically translated in order to convert the file in the format for FlashSim.
- After the conversion, the FlashSim simulator was used to analyze and study the behavior of different workloads, as well as to study different designs of SSDs, e.g. FTL algorithms, wear leveling algorithms, garbage collection policy, etc.

Conclusion

- In this work, an enhanced FlashSim simulator framework and toolkit was designed and prototyped.
- The enhancement focuses on automating trace collection and conversion for the FlashSim simulations.
- The automatic trace collection leverages the *blktrace* utility to collect block I/O traffics.
- A converter used to translate traces for the FlashSim format
- The designed FlashSim extension provides with ease of access to simulations with desired workloads.
- Both benchmark and online streaming application workloads are evaluated to verify the feasibility.
- This study enhances simulation functionality and promotes research on flash memory based storage devices.
- Future research and development include detailed filter/analysis utility on traces.

References

- Chung, T. S., Park, D. J., Park, S., Lee, D. H., Lee, S. W., & Song, H. J. (2009). A survey of flash translation layer. Journal of Systems Architecture, 55(5), 332-343.
- Chen, F., Koufaty, D. A., & Zhang, X. (2009, June). Understanding intrinsic characteristics and system implications of flash memory based solid state drives. In ACM SIGMETRICS Performance Evaluation Review (Vol. 37, No. 1, pp. 181-192). ACM. I.S.
- "blktrace User Guide" http://www.cse.unsw.edu.au/~aaronc/iosched/doc/blktrace.html

Acknowledgement

This research is supported in part by the Edward Whitacre Jr. College of Engineering undergraduate research program and the National Science Foundation under grant CNS-1162488 .

Contact

Roddriguez_80863@students.pupr.edu
wei.xie@ttu.edu
yong.chen@ttu.edu