In the Name of God, the Most Compassionate, the Most Merciful

The DiskSim Simulation Environment Quick Guide

Data Storage Systems and Networks Labratory

Sharif University of Technology

http://dsn.ce.sharif.edu

Alireza Haghdoost

MS Student in Sharif University of Technology

http://ce.sharif.edu/~haghdoost

Agenda

- In this guide we present:
  - DiskSim capabilities
  - SSD Extention
  - Install DiskSim
  - Run DiskSim
  - Configure DiskSim
  - DiskSim Validation results

What is DiskSim

- Accurate, Highly-Configurable Storage Subsystem Simulator
- Developed in Parrallel Data Labratory, Carnegie Mellon
- Capabilities:
  - Simulate a hierarchy of storage components such as buses and controllers (e.g. RAID arrays) as well as disks
  - Using for perfomance evaluation
  - Can be integredet into full system simulators as a disk model
  - Model performance behaviour, but not actual data for each request.

SSD Extension

- Developed by Microsoft Research
- Provide limited support for solid-state-disk (SSD) simulation.
- Not a simulator for any specific SSD, but rather a simulator for an idealized and parameterized SSD (was not Validated)
- Patches over DiskSim
- Patched version Available in the DSN Lab. resource directory.

Installation

- DsikSim delevoped in Linux environment
- Visual Studio port availabe by Microsoft Research
- Ready to make DiskSim is available in resources directory
- Make sure that you have installed build-essential packages
- Just need a make to install
- Kdevelop 3.5 project file also available in the package

Directory Structure

```
~/disksim-4.0$ tree -d -L 1
    |-- diskmodel
    |-- doc
    |-- include
    |-- lib
    |-- libddbg
    |-- libparam
    |-- memsmodel
    |-- src
    |-- ssdmodel
    |-- utils
    |-- valid
    |-- w32build
```

Run DiskSim

```
disksim <parfile> <outfile> <tracetype> <tracefile> <synthgen> [p.o]
```

- `<parfile>` : parameter file
- `<outfile>` : output file (redirect ro terminal by ”stdout”)
- `<tracetype>` : format of input trace file
- `<tracefile>` : trace file used as inuput
- `<synthgen>` : activate synthetic workload generator
- `[p.o]` : allow parfile parameter override

- **Example :**

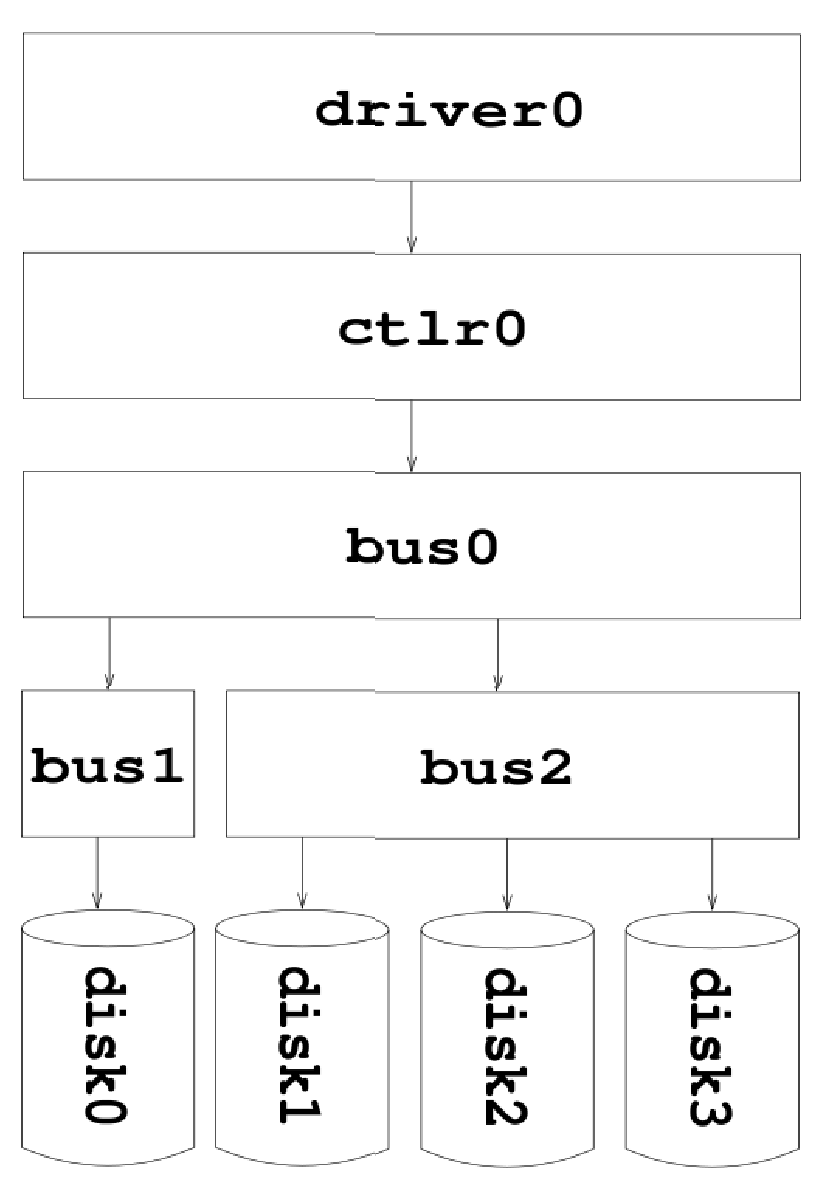
```
$./disksim ssd-sr250k.parv ssd-sr250k.outv ascii 0 1
```

Parameters

- Global
  - Warm-up Ios
  - Init Seed
  - ...
- I/O Simulator
  - I/O trace time scale
  - Location and size scale and offset
  - ...

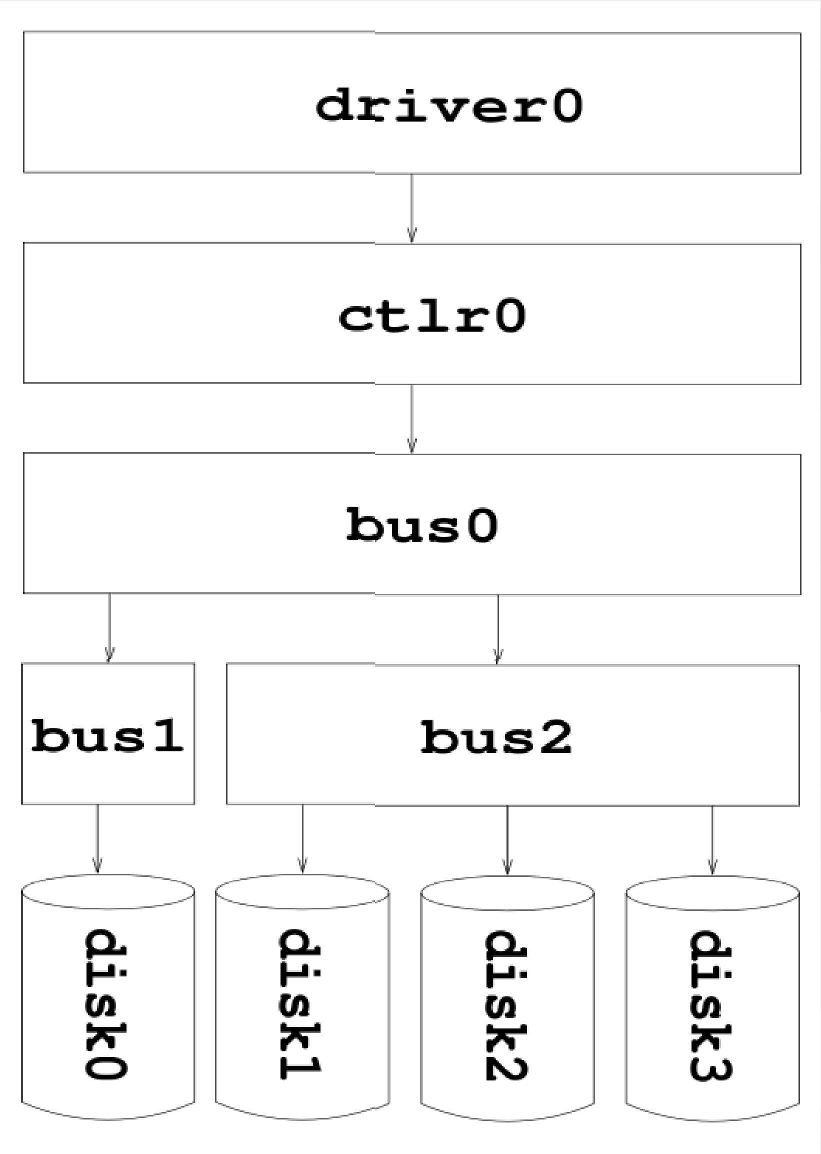
Parameters

- 5 main components of I/O Subsystem:
  - Device Driver
  - Controller
  - Bus
  - Storage Device
  - 2 sub-components:
    - Queue/Scheduler
    - Cache



Parameters

```
# system topology
topology disksim_iodriver driver0 [
   disksim_bus bus0 [
      disksim_ctlr ctlr0 [
         disksim_bus bus1 [
            disksim_disk disk0 []
         ] # end of bus1
         disksim_bus bus2 [
            disksim_disk disk1 []
            disksim_disk disk2 []
            disksim_disk disk3 []
         ] # end of bus2
      ] # end of ctlr0
   ] # end of bus0
] # end of system topology
```



- More topology examples (RAID5) availble in valid directory

Input Wrokloads

- Traces

```
W Hit 30953 54 17003.750000 12394.170572

R Hit 1879119 12 10395.750000 9725.032928

W Hit 288703 8 10858.750000 9405.356666

...
```

- Synthetic Workloads
  - Highly configurable
  - Multiple generator
  - Each synthetic execute as a process

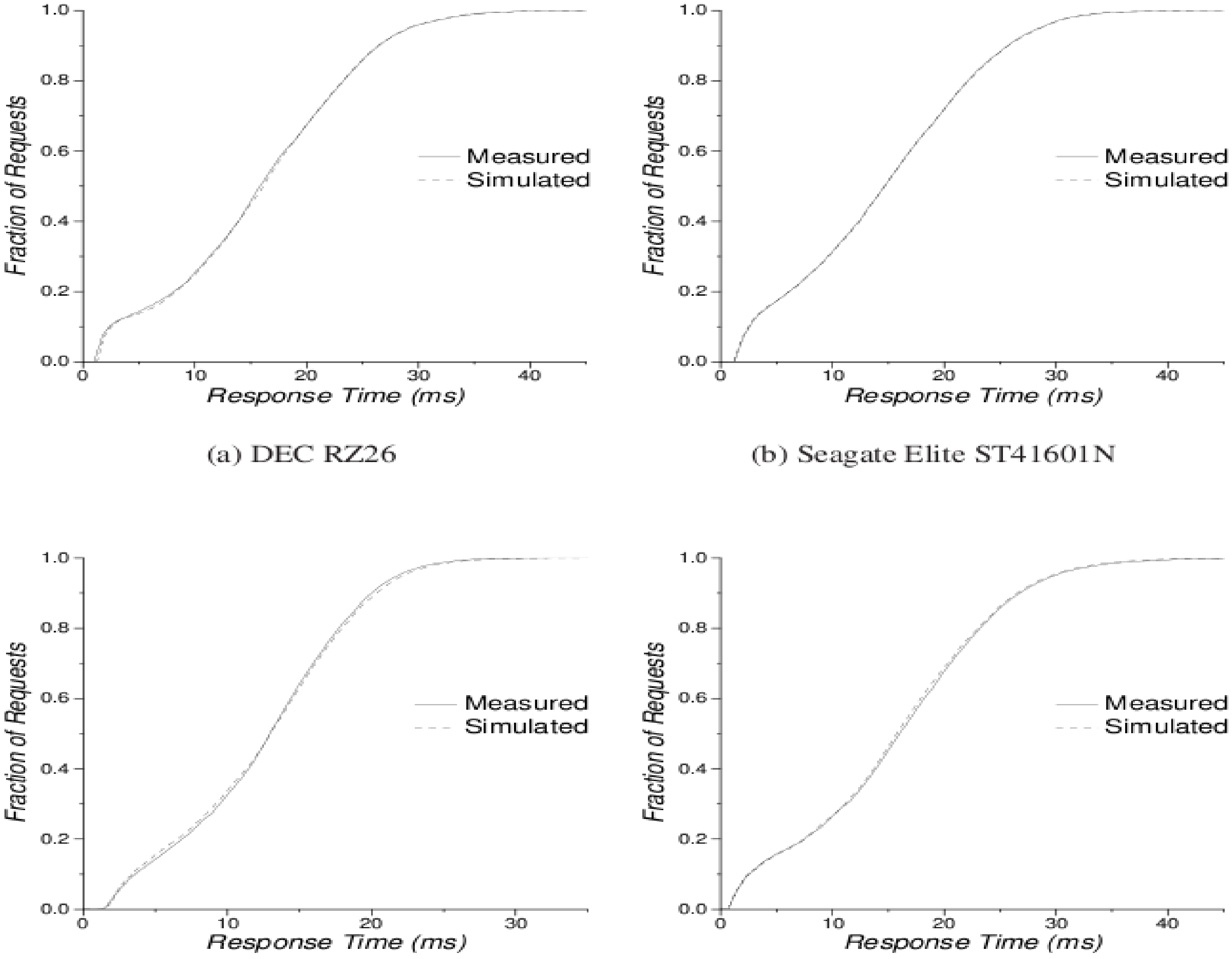
Simulation Results

- Collection of Statistical Results
  - Logical Organization Statistics
    - #read/write
    - #seq. read/write
  - I/O Driver Statistics
    - Idle time
    - Response time
  - Disk/SSD Statistics
    - Idle time
    - Response time
    - IOPS

Simulation Results

- Bus Statistics
    - Utilization time
    - #arbitrations
- Controller Statistics
    - Report disk cache subcomponent statistics
    - #misses/hits
    - #destages

Validation



(a) DEC RZ26

(b) Seagate Elite ST41601N

(c) HP C2490A

(d) HP C3323A