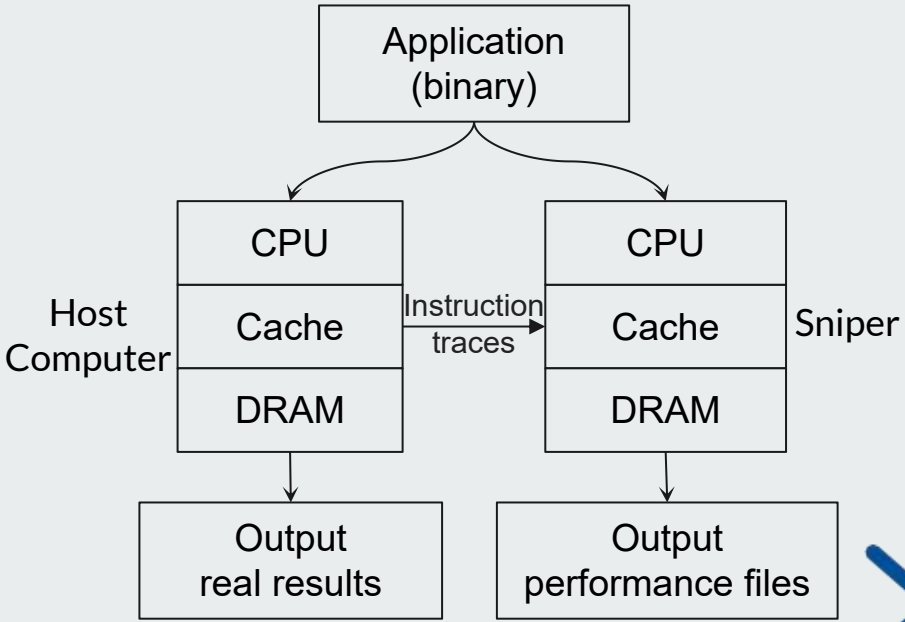
# Sniper Introduction

Chunyan Rong (荣春燕)
rongchy@shanghaitech.edu.cn

# What is Sniper

Sniper is a next generation parallel, high-speed and accurate x86 simulator. The Sniper simulator allows one to perform timing simulations for both multi-program workloads and multi-threaded, shared-memory applications with 10s to 100+ cores.




sniper

Design tomorrow's processor using today's hardware.

# Installation

**Prerequisites:**

OS: Ubuntu 16.04, 18.04, or 20.04

gcc: 7.5.0 or 9.3.0

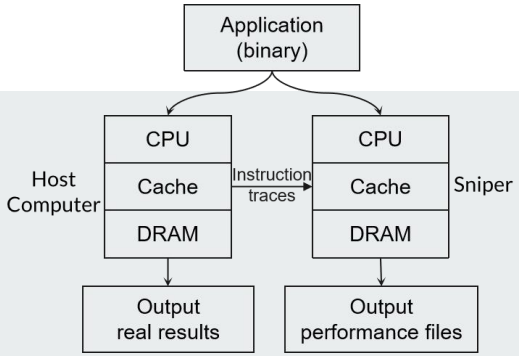
**Installation:**

1. Get the sniper source code from here: Link
2. Input these commands in terminal

```
$ cd sniper
$ sudo dpkg --add-architecture i386
$ sudo apt-get install binutils build-essential curl git libboost-dev libbz2-dev
libc6:i386 libncurses5:i386 libsqlite3-dev libstdc++6:i386 python wget zlib1g-dev
$ chmod -R +x *
$ make                          //compile the code
$ cd test/api; make run    //test
```

Note: use python2, if there are errors about python, check the version of python.

# Test




```
$ cd $(Sniper_Home)/test/api
$ make run
```

../../run-sniper -n 2 -v -c gainestown --roi -g --api/simple/cores_per_socket=1 -- ./api

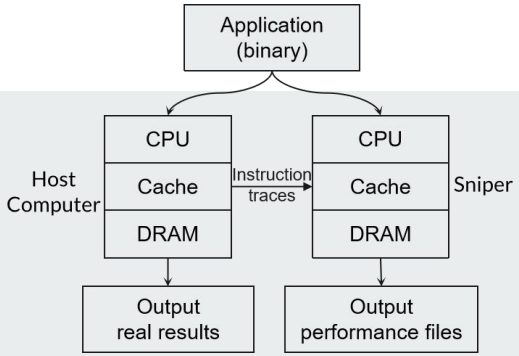
Generate sift file
Line 627

```
$(Sniper_Home)/record-trace
-o /tmp/tmpDJAm3g/run_benchmarks
-v --roi -e 1 -s 0 -r 1 --follow --routine
-tracing -- ./api
```

Main simulation process
Line 787

```
$(Sniper_Home)/lib/sniper -c $(Sniper_Home)/config/base.cfg
 --general/total_cores=2
--general/output_dir=$(Sniper_Home)/test/api
--config=$(Sniper_Home)/config/nehalem.cfg
--config=$(Sniper_Home)/config/gainestown.cfg
--api/simple/cores_per_socket=1 -g --general/magic=true
-g --traceinput/stop_with_first_app=true
-g --traceinput/restart_apps=false
--traceinput/trace_prefix=/tmp/tmpDJAm3g/run_benchmarks
```

# Sift File




```
$ ./record-trace -o api  --  test/api/api -p1
$ ./sift/siftdump api.sift
```

Assembly code

Decoder
(./xed ./xed_kit)

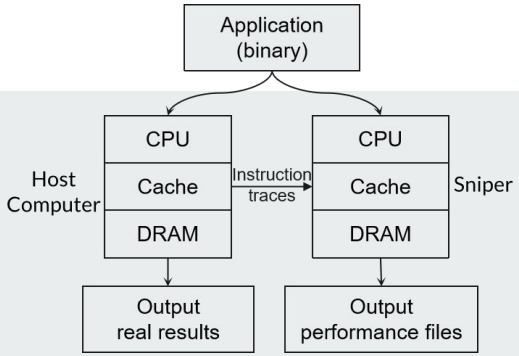
| | | |
|---|---|---|
| 000000000040fa1a | c3 | ret |
| 0000000000400a61 | 89 d0 | mov eax, edx |
| 0000000000400a69 | 48 c7 45 f0 02 00 00 00 | mov qword ptr [rbp-0x10], 0x2 |

/xed/obj/xed-iclass-enum.h (1571)
/xed/obj/xed-iclass-string.c (142)

Multi-programmed workloads

sniper

# Config File



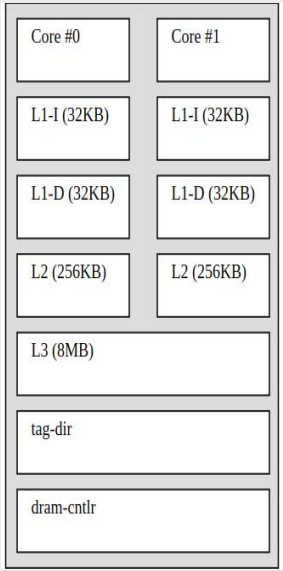

newer values override older ones.

```
$(Sniper_Home)/lib/sniper
-c $(Sniper_Home)/config/base.cfg --
config=$(Sniper_Home)/config/nehalem.cfg --
config=$(Sniper_Home)/config/gainestown.cfg
-g --general/magic=true
```

Auto added ./run-sniper line 716

Added by user

Sim.cfg




[Sniper configuration]

# Output Files

sim.cfg
    Contains the complete configuration details used in this run (./common/system/simulator.cc Line 184)
sim.info
    Record some informations of this run (./run_sniper Line 922-938)
sim.out
    Record some simulation data of this run (./run_sniper Line 919)
sim.stats.sqlite3
    Database file (using the scripts under tool folder parse)
sim_0.log, sim_1.log, app_0.log, system.log
    Log files (log/enabled = true)

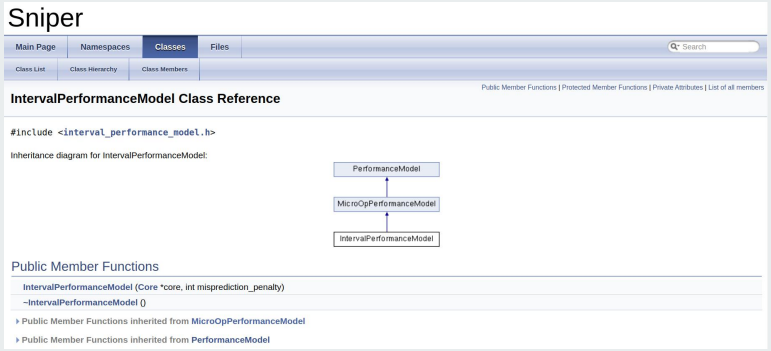
# Folder Structure

- common
  - config ---------- parse config files
  - core ---------- functional and performance models for the core
  - fault_injection ---------- fault injection
  - performance_model ---------- perform the dynamic instructions
  - system ---------- main functional class for simulation
  - trace_frontend ---------- the frontend to process instructions
- config ---------- config files
- xed ---------- instruction decoder
- standalone ---------- standalone.cc (Line10): main function entry
- test ---------- some applications for simulation
- tools ---------- parse output files
- record-trace ---------- generate sift file
- run-sniper ---------- simulator entry

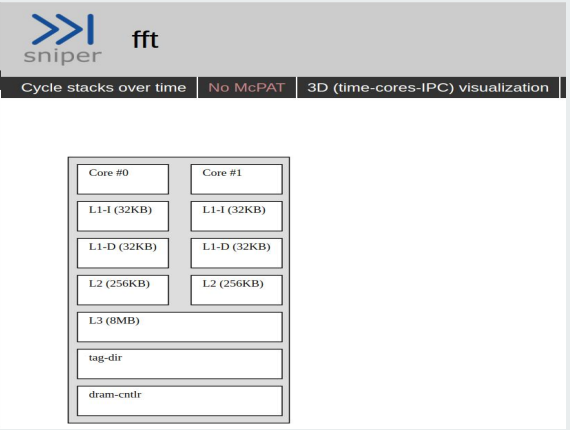
sniper

# Useful Tools

Doxygen:




Viz:


sniper

# Reference Material

Main Page Of Sniper:

https://snipersim.org/w/The_Sniper_Multi-Core_Simulator

Download Source Code:

http://pan.shanghaitech.edu.cn/cloudservice/outerLink/decode?c3Vnb24xNTk5MjA3MTczMzA5c3Vnb24=

Download Sniper manual:

https://snipersim.org/w/Manual

# Thanks