

```

/*****
* LaCASA Laboratory
* Authors: Aleksandar Milenkovic with help of Mounika Ponugoti
* Email: milenkovic@computer.org
* Date: Spring 2016
* Version: 1.0
*****/

```

Getting Started with Parsec-3.0 @LaCASA

1 About Parsec benchmarks

The Princeton Application Repository for Shared-Memory Computer (PARSEC) is a benchmark suite composed of multithreaded programs. The suite focuses on emerging workloads and was designed to be representative of next-generation shared-memory programs for chip-multiprocessors (not focused on HPC workloads). The suite is primarily intended for research. It can also be used for performance measurements of real machines, but its original purpose is insight, not numbers.

2 Parsec @ UAHuntsville

2.1 Location

Machine: eb136i-t1600
 OS: Ubuntu
 Directory: /opt/parsec-3.0; /opt/parsec-2.1;

2.2 Getting Started

Read README file.

The directory structure of the PARSEC distribution is as follows:

```

bin/          Directory with PARSEC tools
config/      Global configuration files
log/         Log files of builds and runs (dynamically created)
man/         Man pages of the PARSEC distribution
pkgs/        Directory containing all the groups which contain all
their packages
  apps/       PARSEC workloads that are applications
  kernels/    PARSEC workloads that are kernels
  libs/       Libraries required by other packages
  tools/      Tools required by other packages
  ...         Additional groups that might have been created

```

Packages have the following directory structure:

```

inputs/
  Directory with input archives (optional)

```

```
inst/ Installation directory (dynamically created)
obj/  Build directory (dynamically created)
```

2.3 Workloads

PARSEC 3.0 contains the original 13 workloads, 3 new network workloads and 2 related benchmark suites i.e., the original SPLASH-2 and the inputs-enlarged SPLASH-2x.

The original 13 workloads:

1. blackscholes Option pricing with Black-Scholes Partial Differential Equation (PDE)
2. bodytrack Body tracking of a person
3. canneal Simulated cache-aware annealing to optimize routing cost of a chip design
4. dedup Next-generation compression with data deduplication
5. facesim Simulates the motions of a human face
6. ferret Content similarity search server
7. fluidanimate Fluid dynamics for animation purposes with Smoothed Particle Hydrodynamics (SPH) method
8. freqmine Frequent itemset mining
9. raytrace Real-time raytracing
10. streamcluster Online clustering of an input stream
11. swaptions Pricing of a portfolio of swaptions
12. vips Image processing
13. x264 H.264 video encoding

3 new network workloads:

1. netdedup Single connection, high bandwidth, large packet size, and latency insensitive (from dedup).
2. netferret Multiple connections, low bandwidth, small packet size, and latency sensitive (from ferret).
3. netstreamcluster Multiple connections, high bandwidth, large packet and latency insensitive (from streamcluster).

2 related benchmark suites:

1. SPLASH-2

Recent studies show that SPLASH-2 and PARSEC benchmark suites complement each other well in terms of diversity of architectural characteristics such as instruction distribution, cache miss rate and working set size. In order to provide computer architects with the convenient use of both benchmarks, we have integrated SPLASH-2 into the PARSEC environment in this release. Users can now build, run and manage both workloads under the same environment framework.

2. SPLASH-2x

SPLASH-2x has several input datasets at different scales. To scale up the input sets for SPLASH-2, we have explored the input space of the SPLASH-2 workloads to generate multiple scales of input sets, similar to PARSEC's criterion: native, simlarge, simmedium, simsmall.

2.4 Inputs

Parsec provides 6 different sizes of inputs for each benchmark. Those are test, simdev, simsmall, simmedium, simlarge, and native.

3 Running Parsec benchmarks from command lines

We often want to run individual benchmarks from a command line or to run a benchmark on top of an architectural simulator. To do this, you may locate executables and input files and copy them in your working directory. You will also need to learn how to launch an individual benchmark. The following command lines show how to launch each benchmark without using parsecmgmt tool.

blacksholes

```
native input:
    /opt/parsec-3.0/pkgs/apps/blacksholes/inst/amd64-linux.gcc/bin/blacksholes
    ${NTHREADS} in_10M.txt prices.txt
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/blacksholes/inst/amd64-linux.gcc/bin/blacksholes
    ${NTHREADS} in_16.txt prices.txt
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/blacksholes/inst/amd64-linux.gcc/bin/blacksholes
    ${NTHREADS} in_64K.txt prices.txt
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/blacksholes/inst/amd64-linux.gcc/bin/blacksholes
    ${NTHREADS} in_16K.txt prices.txt
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/blacksholes/inst/amd64-linux.gcc/bin/blacksholes
    ${NTHREADS} in_4K.txt prices.txt
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/blacksholes/inst/amd64-linux.gcc/bin/blacksholes
    ${NTHREADS} in_4.txt prices.txt
```

bodytrack

```
native input:
    /opt/parsec-3.0/pkgs/apps/bodytrack/inst/amd64-linux.gcc/bin/bodytrack
    sequenceB_261 4 261 4000 5 0 ${NTHREADS}
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/bodytrack/inst/amd64-linux.gcc/bin/bodytrack sequenceB_1
    4 1 100 3 0 ${NTHREADS}
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/bodytrack/inst/amd64-linux.gcc/bin/bodytrack sequenceB_4
    4 4 4000 5 0 ${NTHREADS}
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/bodytrack/inst/amd64-linux.gcc/bin/bodytrack sequenceB_2
    4 2 2000 5 0 ${NTHREADS}
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/bodytrack/inst/amd64-linux.gcc/bin/bodytrack sequenceB_1
    4 1 1000 5 0 ${NTHREADS}
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/bodytrack/inst/amd64-linux.gcc/bin/bodytrack sequenceB_1
    4 1 5 1 0 ${NTHREADS}
```

canneal

```
native input:
    /opt/parsec-3.0/pkgs/apps/canneal/inst/amd64-linux.gcc/bin/canneal ${NTHREADS}
    15000 2000 2500000.nets 6000
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/canneal/inst/amd64-linux.gcc/bin/canneal ${NTHREADS} 100
    300 100.nets 2
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/canreal/inst/amd64-linux.gcc/bin/canreal ${NTHREADS}
15000 2000 400000.nets 128
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/canreal/inst/amd64-linux.gcc/bin/canreal ${NTHREADS}
15000 2000 200000.nets 64
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/canreal/inst/amd64-linux.gcc/bin/canreal ${NTHREADS}
10000 2000 100000.nets 32
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/canreal/inst/amd64-linux.gcc/bin/canreal ${NTHREADS} 5
100 10.nets 1
```

dedup

```
native input:
    /opt/parsec-3.0/pkgs/apps/dedup/inst/amd64-linux.gcc/bin/dedup -c -p -v -t
${NTHREADS} -i FC-6-x86_64-disc1.iso -o output.dat.ddp
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/dedup/inst/amd64-linux.gcc/bin/dedup -c -p -v -t
${NTHREADS} -i hamlet.dat -o output.dat.ddp
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/dedup/inst/amd64-linux.gcc/bin/dedup -c -p -v -t
${NTHREADS} -i media.dat -o output.dat.ddp
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/dedup/inst/amd64-linux.gcc/bin/dedup -c -p -v -t
${NTHREADS} -i media.dat -o output.dat.ddp
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/dedup/inst/amd64-linux.gcc/bin/dedup -c -p -v -t
${NTHREADS} -i media.dat -o output.dat.ddp
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/dedup/inst/amd64-linux.gcc/bin/dedup -c -p -v -t
${NTHREADS} -i test.dat -o output.dat.ddp
```

facesim

```
native input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/facesim -timing -threads
${NTHREADS} -lastframe 100
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/facesim -timing -threads
${NTHREADS}
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/facesim -timing -threads
${NTHREADS}
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/facesim -timing -threads
${NTHREADS}
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/facesim -timing -threads
${NTHREADS}
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/facesim -h
```

ferret

```
native input:
    /opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/ferret core1 lsh queries
50 20 ${NTHREADS} output.txt
```

```
simdev input:
/opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/ferret corel lsh queries
5 5 ${NTHREADS} output.txt
```

```
simlarge input:
/opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/ferret corel lsh queries
10 20 ${NTHREADS} output.txt
```

```
simmedium input:
/opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/ferret corel lsh queries
10 20 ${NTHREADS} output.txt
```

```
simsmall input:
/opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/ferret corel lsh queries
10 20 ${NTHREADS} output.txt
```

```
test input:
/opt/parsec-3.0/pkgs/apps/facesim/inst/amd64-linux.gcc/bin/ferret corel lsh queries
5 5 ${NTHREADS} output.txt
```

fluidanimate

```
native input:
/opt/parsec-3.0/pkgs/apps/fluidanimate/inst/amd64-linux.gcc/bin/fluidanimate
${NTHREADS} 500 in_500K.fluid out.fluid
```

```
simdev input:
/opt/parsec-3.0/pkgs/apps/fluidanimate/inst/amd64-linux.gcc/bin/fluidanimate
${NTHREADS} 3 in_15K.fluid out.fluid
```

```
simlarge input:
/opt/parsec-3.0/pkgs/apps/fluidanimate/inst/amd64-linux.gcc/bin/fluidanimate
${NTHREADS} 5 in_300K.fluid out.fluid
```

```
simmedium input:
/opt/parsec-3.0/pkgs/apps/fluidanimate/inst/amd64-linux.gcc/bin/fluidanimate
${NTHREADS} 5 in_100K.fluid out.fluid
```

```
simsmall input:
/opt/parsec-3.0/pkgs/apps/fluidanimate/inst/amd64-linux.gcc/bin/fluidanimate
${NTHREADS} 5 in_35K.fluid out.fluid
```

```
test input:
/opt/parsec-3.0/pkgs/apps/fluidanimate/inst/amd64-linux.gcc/bin/fluidanimate
${NTHREADS} 1 in_5K.fluid out.fluid
```

freqmine

```
native input:
export OMP_NUM_THREADS=${NTHREADS}
/opt/parsec-3.0/pkgs/apps/freqmine/inst/amd64-linux.gcc/bin/freqmine
webdocs_250k.dat 11000
```

```
simdev input:
export OMP_NUM_THREADS=${NTHREADS}
/opt/parsec-3.0/pkgs/apps/freqmine/inst/amd64-linux.gcc/bin/freqmine
T10I4D100K_1k.dat 3
```

```
simlarge input:
export OMP_NUM_THREADS=${NTHREADS}
/opt/parsec-3.0/pkgs/apps/freqmine/inst/amd64-linux.gcc/bin/freqmine
kosarak_990k.dat 790
```

```
simmedium input:
export OMP_NUM_THREADS=${NTHREADS}
/opt/parsec-3.0/pkgs/apps/freqmine/inst/amd64-linux.gcc/bin/freqmine
kosarak_500k.dat 410
```

```
simsmall input:
export OMP_NUM_THREADS=${NTHREADS}
/opt/parsec-3.0/pkgs/apps/freqmine/inst/amd64-linux.gcc/bin/freqmine
kosarak_250k.dat 220
```

```
test input:
    export OMP_NUM_THREADS=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/freqmine/inst/amd64-linux.gcc/bin/freqmine
T10I4D100K_3.dat 1
```

raytrace

```
native input:
    /opt/parsec-3.0/pkgs/apps/raytrace/inst/amd64-linux.gcc/bin/rtview thai_statue.obj
-automove -nthreads ${NTHREADS} -frames 200 -res 1920 1080
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/raytrace/inst/amd64-linux.gcc/bin/rtview bunny.obj -
automove -nthreads ${NTHREADS} -frames 1 -res 16 16
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/raytrace/inst/amd64-linux.gcc/bin/rtview happy_buddha.obj
-automove -nthreads ${NTHREADS} -frames 3 -res 1920 1080
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/raytrace/inst/amd64-linux.gcc/bin/rtview happy_buddha.obj
-automove -nthreads ${NTHREADS} -frames 3 -res 960 540
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/raytrace/inst/amd64-linux.gcc/bin/rtview happy_buddha.obj
-automove -nthreads ${NTHREADS} -frames 3 -res 480 270
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/raytrace/inst/amd64-linux.gcc/bin/rtview octahedron.obj -
automove -nthreads ${NTHREADS} -frames 1 -res 1 1
```

streamcluster

```
native input:
    /opt/parsec-3.0/pkgs/apps/streamcluster/inst/amd64-linux.gcc/bin/streamcluster 10
20 128 1000000 200000 5000 none output.txt ${NTHREADS}
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/streamcluster/inst/amd64-linux.gcc/bin/streamcluster 3 10
3 16 16 10 none output.txt ${NTHREADS}
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/streamcluster/inst/amd64-linux.gcc/bin/streamcluster 10
20 128 16384 16384 1000 none output.txt ${NTHREADS}
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/streamcluster/inst/amd64-linux.gcc/bin/streamcluster 10
20 64 8192 8192 1000 none output.txt ${NTHREADS}
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/streamcluster/inst/amd64-linux.gcc/bin/streamcluster 10
20 32 4096 4096 1000 none output.txt ${NTHREADS}
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/streamcluster/inst/amd64-linux.gcc/bin/streamcluster 2 5
1 10 10 5 none output.txt ${NTHREADS}
```

swaptions

```
native input:
    /opt/parsec-3.0/pkgs/apps/swaptions/inst/amd64-linux.gcc/bin/swaptions -ns 128 -sm
1000000 -nt ${NTHREADS}
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/swaptions/inst/amd64-linux.gcc/bin/swaptions -ns 3 -sm 50
-nt ${NTHREADS}
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/swaptions/inst/amd64-linux.gcc/bin/swaptions -ns 64 -sm
40000 -nt ${NTHREADS}
```

```
simmedium input:
```

```
    /opt/parsec-3.0/pkgs/apps/swaptions/inst/amd64-linux.gcc/bin/swaptions -ns 32 -sm
20000 -nt ${NTHREADS}
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/swaptions/inst/amd64-linux.gcc/bin/swaptions -ns 16 -sm
10000 -nt ${NTHREADS}
```

```
test input:
    /opt/parsec-3.0/pkgs/apps/swaptions/inst/amd64-linux.gcc/bin/swaptions -ns 1 -sm 5
-nt ${NTHREADS}
```

vips

```
native input:
    export IM_CONCURRENCY=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/vips/inst/amd64-linux.gcc/bin/vips im_benchmark
orion_18000x18000.v output.v
```

```
simdev input:
    export IM_CONCURRENCY=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/vips/inst/amd64-linux.gcc/bin/vips im_benchmark
barbados_256x288.v output.v
```

```
simlarge input:
    export IM_CONCURRENCY=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/vips/inst/amd64-linux.gcc/bin/vips im_benchmark
bigben_2662x5500.v output.v
```

```
simmedium input:
    export IM_CONCURRENCY=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/vips/inst/amd64-linux.gcc/bin/vips im_benchmark
vulture_2336x2336.v output.v
```

```
simsmall input:
    export IM_CONCURRENCY=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/vips/inst/amd64-linux.gcc/bin/vips im_benchmark
pomegranate_1600x1200.v output.v
```

```
test input:
    export IM_CONCURRENCY=${NTHREADS}
    /opt/parsec-3.0/pkgs/apps/vips/inst/amd64-linux.gcc/bin/vips im_benchmark
barbados_256x288.v output.v
```

x264

```
native input:
    /opt/parsec-3.0/pkgs/apps/x264/inst/amd64-linux.gcc/bin/x264 --quiet --qp 20 --
partitions b8x8,i4x4 --ref 5 --direct auto --b-pyramid --weightb --mixed-refs --no-fast-
pskip --me umh --subme 7 --analyse b8x8,i4x4 --threads ${NTHREADS} -o eledream.264
eledream_1920x1080_512.y4m
```

```
simdev input:
    /opt/parsec-3.0/pkgs/apps/x264/inst/amd64-linux.gcc/bin/x264 --quiet --qp 20 --
partitions b8x8,i4x4 --ref 5 --direct auto --b-pyramid --weightb --mixed-refs --no-fast-
pskip --me umh --subme 7 --analyse b8x8,i4x4 --threads ${NTHREADS} -o eledream.264
eledream_64x36_3.y4m
```

```
simlarge input:
    /opt/parsec-3.0/pkgs/apps/x264/inst/amd64-linux.gcc/bin/x264 --quiet --qp 20 --
partitions b8x8,i4x4 --ref 5 --direct auto --b-pyramid --weightb --mixed-refs --no-fast-
pskip --me umh --subme 7 --analyse b8x8,i4x4 --threads ${NTHREADS} -o eledream.264
eledream_640x360_128.y4m
```

```
simmedium input:
    /opt/parsec-3.0/pkgs/apps/x264/inst/amd64-linux.gcc/bin/x264 --quiet --qp 20 --
partitions b8x8,i4x4 --ref 5 --direct auto --b-pyramid --weightb --mixed-refs --no-fast-
pskip --me umh --subme 7 --analyse b8x8,i4x4 --threads ${NTHREADS} -o eledream.264
eledream_640x360_32.y4m
```

```
simsmall input:
    /opt/parsec-3.0/pkgs/apps/x264/inst/amd64-linux.gcc/bin/x264 --quiet --qp 20 --
partitions b8x8,i4x4 --ref 5 --direct auto --b-pyramid --weightb --mixed-refs --no-fast-
```



```
pskip --me umh --subme 7 --analyse b8x8,i4x4 --threads ${NTHREADS} -o eledream.264  
eledream_640x360_8.y4m
```

test input:

```
/opt/parsec-3.0/pkgs/apps/x264/inst/amd64-linux.gcc/bin/x264 --quiet --qp 20 --  
partitions b8x8,i4x4 --ref 5 --direct auto --b-pyramid --weightb --mixed-refs --no-fast-  
pskip --me umh --subme 7 --analyse b8x8,i4x4 --threads ${NTHREADS} -o eledream.264  
eledream_32x18_1.y4m
```

Note: Replace \${NTHREADS} with the number of threads.

