PARSEC (The Princeton Application Repository for Shared-Memory Computers)是一个**多线程**应用程序组成的测试程序集。该程序集代表了未来运行在片上多核系统中的共享内存应用程序的发展趋势。

片上多核处理器已经成为通用处理器的主流。这一转变带来了巨大的效应：在短期时间内，如果不改变底层代码，显著的性能提升是无法实现的。未来应用程序必须做出重大改变—必须改为并行程序。目前，由于并行程序开发和调试的困难较高，软件开发者还没有转为开发并行应用程序，这使得计算机架构师和芯片设计者缺乏具体的未来应用实例，无法进一步设计新的、高效的处理器。

PARSEC的目的就是让未来的应用程序在当前成为现实，即PARSEC中的应用程序代表了未来的应用程序的主流，以给予计算机架构师和芯片设计者应用依据，方便其进一步开发、设计处理器。

## 主要特点：

PARSEC 与其他测试程序的特点（不同点）如下:

多线程（并发性）: 虽然串行程序很多，但是它们限制了多核处理器机器的发展，PARSEC 是为数不多的并发程序的测试集。

新型负载: 该测试集包含刚出现的新型负载程序，这些应用程序虽然未被广泛使用，但却是未来应用的主流方向。PARSEC的目标就是提供在未来几年可能会成为主流应用的测试程序。

多元化: PARSEC并非像之前的一些测试程序仅仅试图开发单一领域的应用程序，在其测试程序集中涉及多个应用领域，并试图选取最具代表性的应用实例。

非针对高性能: 计算密集的并行程序在高性能计算中非常普遍，但是高性能程序仅仅只是应用程序中的一个小分支。在未来并行技术将会普及到各个应用领域。PARSEC测试程序集的开发者并不将并行程序局限于高性能计算，而是涉及到应用的各个领域，从桌面程序到服务器应用。

研究性: 这个测试程序集主要是供研究使用。虽然也可以用来测试实际机器的性能，但是其只是给设计者以启示，而不是给予性能评价的具体分数。

1. 依赖库

1）apt-get install m4;

2）apt-get install g++;

3) apt-get install libxi-dev

apt-get install libxmu-dev

4）<http://www.gnu.org/software/gettext/gettext.html> 下载安装gettext：

$ ./configure

$ make

$ sudo make install

1. 解压

parsec3.0不需要安装，解压开就可以用了



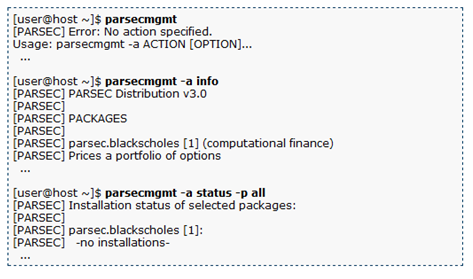
1. 设置环境

进入parsec3.0所在的目录，执行source env.sh



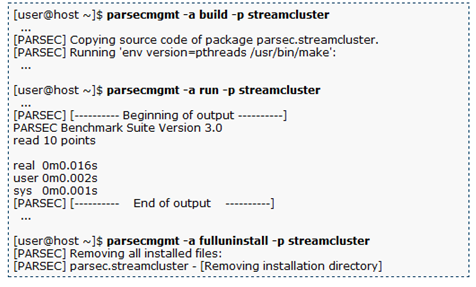
这个在每个新打开的终端都需要执行一次。

然后输入 parsecmgmt

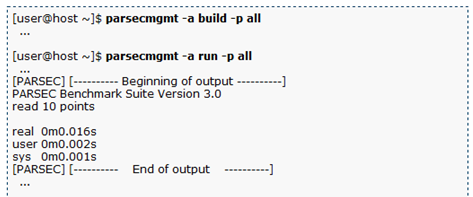


1. 构建/运行/卸载benchmark

以streamcluster为例，



构建并运行所有benchmark



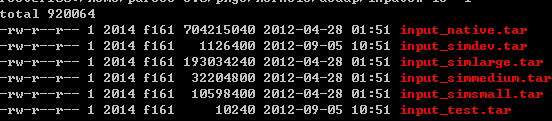
1. 选择其他输入数据集

下载parsec-3.0-input-sim.tar.gz，解压至parsec3.0所在目录

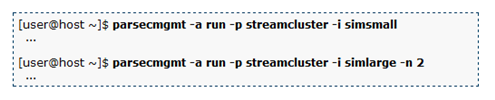


运行程序streamcluster时选择其他数据集

查看parsec3.0目录下streamcluster文件夹下inputs目录：



使用simlarge数据集：



-i 命令即选择数据集，一般实际测试中我们选择native数据集，如果没有-i选项，默认的是使用test数据集；

-n 即运行的线程数目，-n 2即开启2个线程跑。

Usage: parsecmgmt -a ACTION [OPTION]...

Manage the installation of the PARSEC benchmark suite.

Options:

-a ACTION Specifies the action to perform. See below for a list of valid actions.

-p PACKAGE A list of packages or aliases on which the action is to be performed.

-c CONFIG Which build configuration to use. Default: 'gcc'

-d RUNDIR Use directory RUNDIR as root in which to run the benchmarks.

**-i INPUT The input to use to run the benchmarks. Default: 'test'。选择测试数据大小，如native，simlarge等，在对应benchmark目录下input中**

-n THREADS The minimum number of threads to use. Default: '1'

-s SUBMIT Command to use to submit the benchmark for execution.Default: 'time'

-x EXTENSION Extension to append to platform descriptor string.Default: none

-k Keep & use run directory as found, do not unpack inputs for benchmark execution. Assume everything is already set up.

-m NETMODE Execution mode for network workloads. NETMODE can be either 'server' or 'client'.

-t THREADS The number of client threads to connect to the server. Default: '1'

-l Disable log.

-h Displays this help message.

Actions:

'build' Builds and installs the specified packages.

'run' Runs the specified packages.

'clean' Removes all files generated during the build and run phase of the listed packages for the current platform, but leaves the installed files untouched.

'uninstall' Removes the installed files of the listed packages for the current platform.

'fullclean' Like 'clean', but for all architectures.

'fulluninstall' Like 'uninstall', but for all architectures.

'status' Shows a summary of the current status of the PARSEC installation.

'info' List available packages and configurations.

Examples (1) [Compatible to PARSEC 2.1 but Extended]:

- Build all benchmarks of three suites, i.e., PARSEC, SPLASH-2 and SPLASH-2x

:

parsecmgmt -a build -p all

- Build only applications, use 'icc' as build configuration:

parsecmgmt -a build -c icc -p apps

- Do a full cleanup after a build or benchmark run:

parsecmgmt -a fullclean -p all

- Remove all generated files of the current architecture:

parsecmgmt -a uninstall -p all

- Run benchmark 'ferret' w/ input 'simsmall' and 4 threads:

parsecmgmt -a run -p ferret -i simsmall -n 4

- Get a quick summary of all available packages and features:

parsecmgmt -a info

- Show which kernel binaries have been installed:

parsecmgmt -a status -p kernels

Examples (2) [NEW: Network Benchmarks]:

- Check the status of all components involved in network benchmarks:

parsecmgmt -a status -p netapps

- Build network benchmark 'netstreamcluster':

parsecmgmt -a build -p netstreamcluster

- Build all network benchmark:

parsecmgmt -a build -p netapps

## - Run network benchmark 'netdedup' w/ input 'native' and 2 server threads:

parsecmgmt -a run -p netdedup -i native -n 2

- Run network benchmark 'netferret' w/ input 'simlarge', 4 server threads and

2 client connections:

parsecmgmt -a run -p netferret -i simlarge -n 4 -t 2

- For simulation, run 'netdedup' server on a simulator w/ 4 threads and run

client on a real machine:

parsecmgmt -a run -p netdedup -i simlarge -n 4 -m server

parsecmgmt -a run -p netdedup -i simlarge -m client

- Do a full cleanup for network benchmarks:

parsecmgmt -a fullclean -p netapps

Examples (3) [NEW: SPLASH-2 Suite and SPLASH-2x Suite]:

- Check the status of SPLASH-2 suite and SPLASH-2x suite:

parsecmgmt -a status -p splash2

parsecmgmt -a status -p splash2x

- Build benchmark 'raytrace' from SPLASH-2x suite other than PARSEC:

parsecmgmt -a build -p splash2x.raytrace

parsecmgmt -a build -p raytrace ## defaultly from PARSEC (for comparison)

- Build benchmark 'fft' from SPLASH-2 suite with 'gcc-serial' build configuration:

parsecmgmt -a build -c gcc-serial -p splash2.fft

- Build all benchmarks from SPLASH-2 suite and SPLASH-2x suite:

parsecmgmt -a build -p splash2

parsecmgmt -a build -p splash2x

- Run benchmark 'fft' from SPLASH-2x w/ input 'simsmall' and 4 threads:

parsecmgmt -a run -p splash2x.fft -i simsmall -n 4

- Do a full cleanup for SPLASH-2 suite

parsecmgmt -a fullclean -p splash2

|  |  |  |
| --- | --- | --- |
| 编号 | 程序名称 | 描述 |
| s1 | blackscholes | Calculates portfolio price using Black-Scholes PDE |
| s2 | bodytrack | Computer vision, tracks 3D pose of human body |
| s3 | canneal | Synthetic chip design, routing |
| s4 | facesim | Physics simulation, models a human face |
| s5 | ferret | Pipelined audio, image and video searches |
| s6 | fluidanimate | Physics simulation, animation of fluids |
| s7 | freqmine | Data mining application |
| s8 | raytrace | Computer animation application |
| s9 | streamcluster | Kernel to solve the online clustering problem |
| s10 | swaptions | Computes portfolio prices using Monte-Carlo simulation |
| s11 | vips | Image processing, image transformations |
| s12 | x264 | H.264 video encoder |