

How To Develop High Performance Applications

Jaehyuk Lee

System Engineer, SCJD

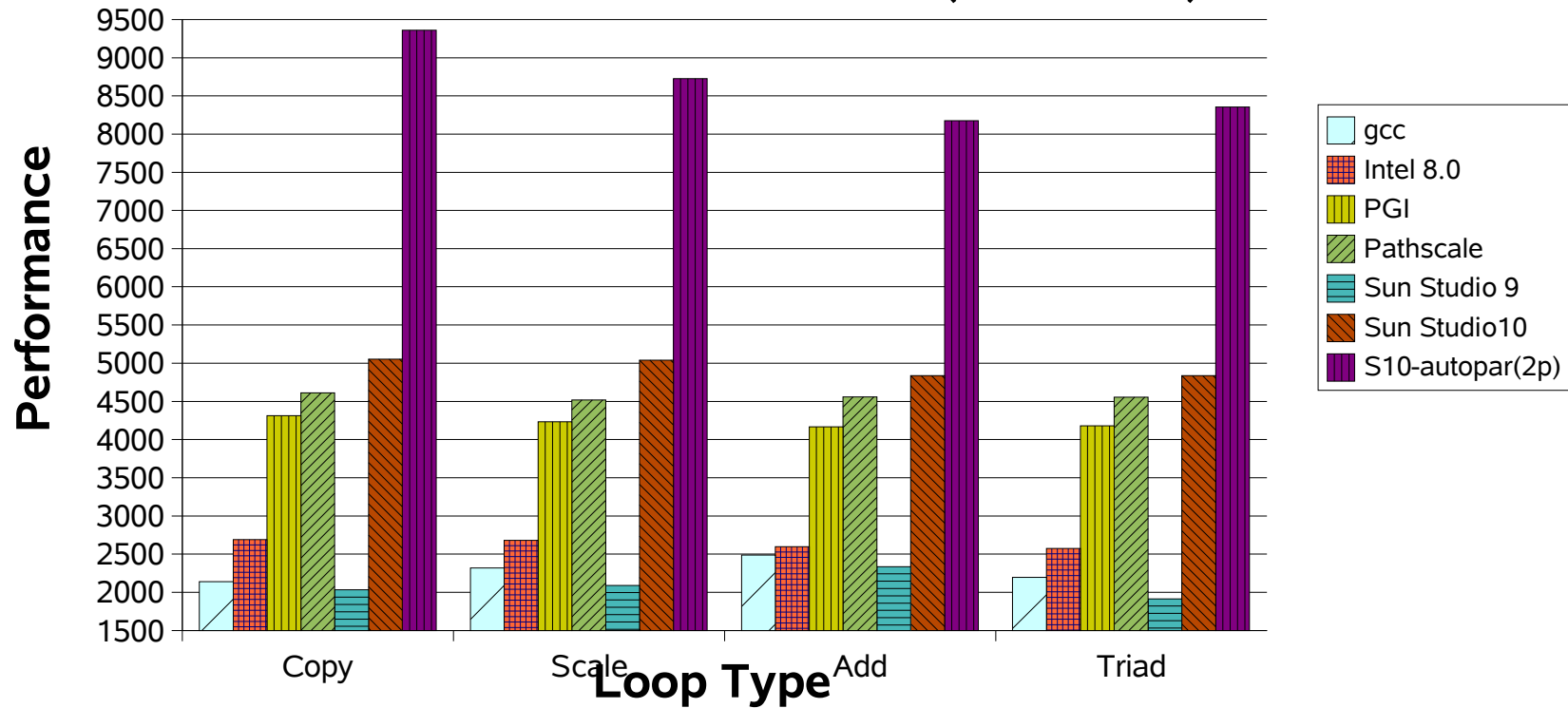
OEM Software Sales

Studio 10 Performance World Records

- x86/x64 Records:
 - 4 World Record SPEC OMPM2001 results(2p,4p, dual-core)
 - World Record SPEC CPU2000 results on SunFire V40z
 - <http://www.sun.com/software/solaris/benchmarks.jsp>
 - J2SE 5.0 compiled with Studio10 claimed 5 SPECjbb2000 records
 - BLAST, on Solaris10 using Studio10, runs 34-61% faster than Dell Precision 320
- SPARC Records:
 - Sun Fire 25K server running Solaris10 delivers highest TPC-H @3000GB
 - Sun Fire 6900 delivered record throughput using Oracle Applications Standard Batch benchmark
 - SunFire 4900 with Solaris10, Oracle 10g delivered first Oracle Apps Batch (HVOP) submission

Studio 10: Ex-STREAM performance!

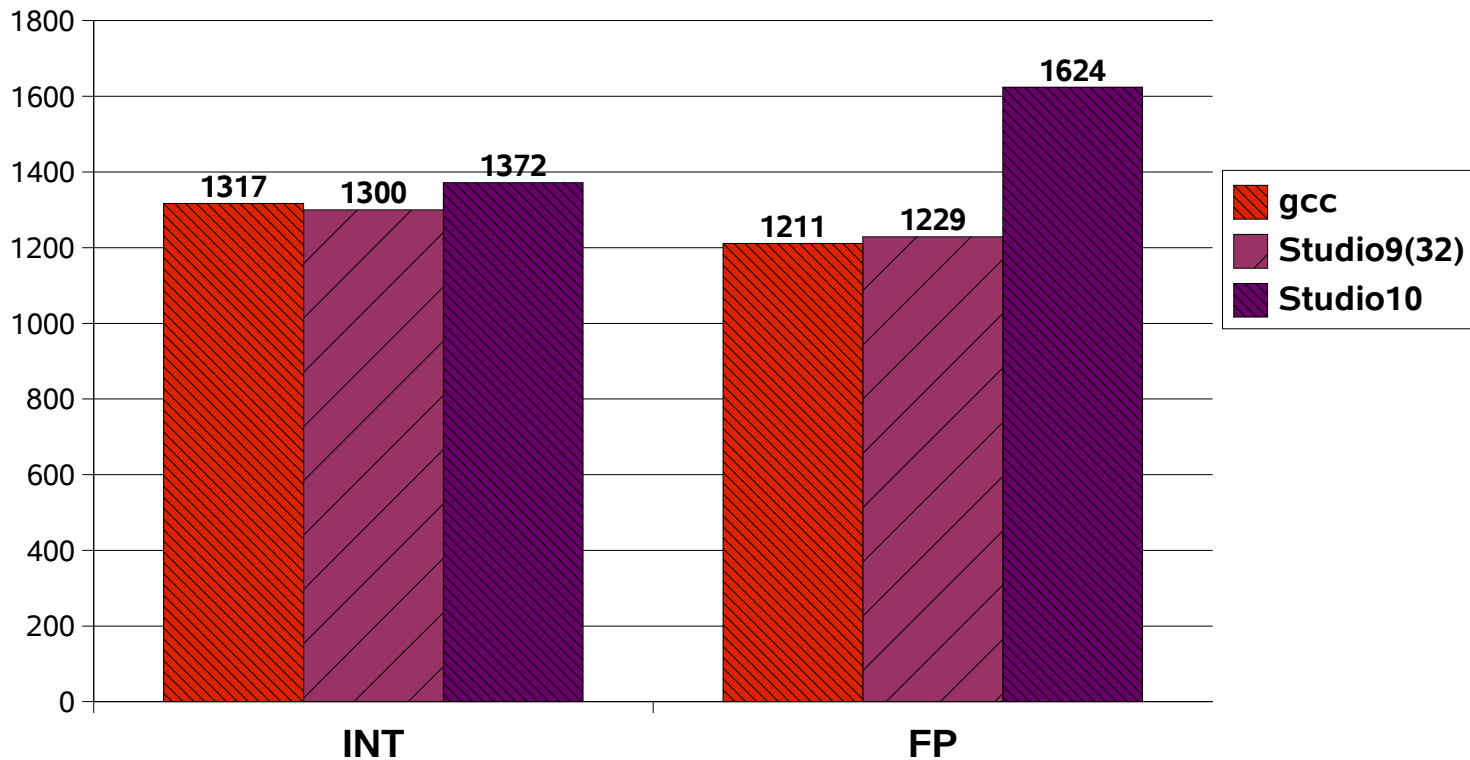
STREAM Benchmark
 All numbers on AMD64 x50 box (2.4GHz)
 Solaris10 or SuSE9 (for Linux)



Improvements due to -xautopar, micro-vectorization and prefetch

Studio 10: SPEC CPU2000 Performance

**AMD64 SPEC CPU2000 (higher is better)
2.4GHz, AMD Opteron x50 Whitebox
All measurements are on Solaris 10**



Compiler Basic Information

- Default location of the compilers: /opt/SUNWspro/bin
- Various sources of information
 - Compiler options (-xhelp = flags and -xhelp=readme)
 - <file:///opt/SUNWspro/docs/index.html>
 - <http://developers.sun.com>
- Use the -V option for compiler release information
- The following macros are set:
 - cc __SUNPRO_C
 - f90/f95 __SUNPRO_F90 and __SUNPROS_F95

Demo Performance Data

- Time to find the primes up to 3,000,000

1) prime_demo_s: 11.12 sec base

> -g

2) prime_demo_f: 7.61 sec 31.6% faster

> -fast

3) prime_demo_v: 7.30 sec 4.1% inc faster

> -fast -xrestrict -xipo

4) prime_demo_p: (2p) 6.29 sec 13.8% inc faster

> -fast -xrestrict -xipo -xopenmp=parallel

5) prime_demo_p:(24p) 0.55 sec 91.3% inc faster

Compiling for performance, Optimization Flags

- **-O** (**=-xO3**)
- **-xO1~5**
 - 1: *basic local optimization*
 - 2: *local & global optimization. minimum size*
 - 3: *Optimizes references or definitions for external variables. Size trade off*
 - 4: *inlining. Size trade off*
 - 5: *Uses optimization algorithms. Can be improved.*
- **-fast**
 - *easy to use, best performance on most code, but it assumes compile platform = run platform and makes FP arithmetic simplification*
- **-xrestrict={%all(dv)|%none(d)|fn[,fn...]}**
- **-xcross=0(d)|1(dv)**
 - *Only effective with xO4 or -xO5, source file base*
- **-xipo={0(d)|1(dv)|2}**
 - *with -xipo=1, the compiler performs inlining across all source files*
 - *include object file, but not libraries.*
 - *the compiler performs interprocedural aliasing analysis as well as optimizations of memory allocation and layout to improve cache performance*
- **-fsimple={0(d)|1(dv)|2}**
 - *FP arithmetic can be simplified*

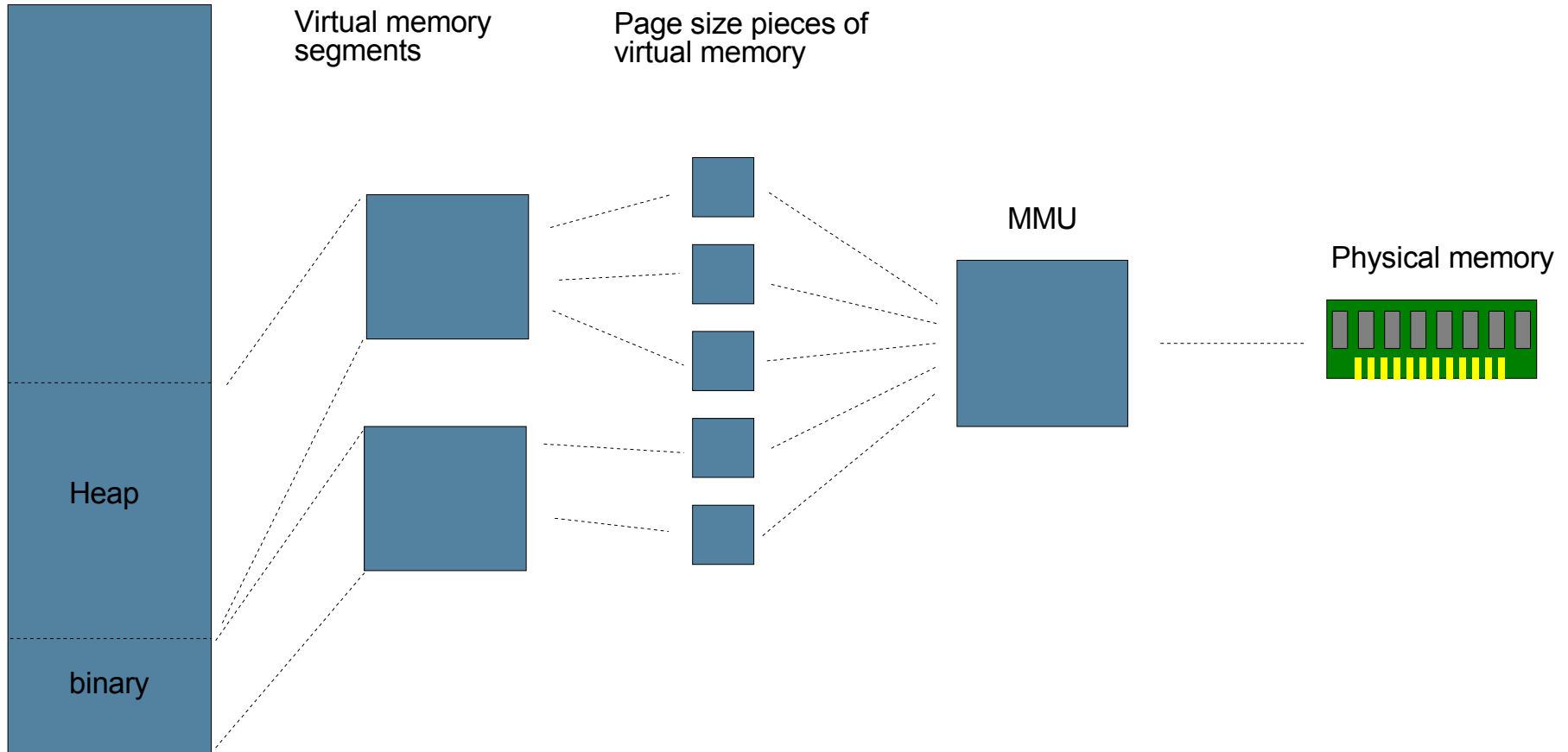
About Math Libraries

- A well-known C math library is called “libm”
 - It is accessed through the -lm link option
- A faster version of libm called libmopt also available
 - Fortran and C++: use the -xlibmopt link option
 - C: Add -lmopt to your link line
- Inline versions of libm are available too
 - Some math functions are inlined at compile time
 - The -xlibmil compiler option can be used
 - Supported on f95, cc and CC compilers

Multiple Page Size Support (MPSS)

Solaris OS Virtual-to-Physical Memory Management

Process's Linear Virtual address space



Memory Management Unit, MMU

- **Translate virtual memory to Physical memory**
- **TLB(translation lookaside buffer)**
 - *In processor*
 - *small size*
 - * *UltraSPARCI/II : 64 entries (64 x 8K)*
- **TSB(translation storage buffer,pagetable)**
 - *in memory*

TLB miss

```
so19# trapstat -T 1
```

cpu	m	size	itlb-miss	%tim	itsb-miss	%tim	dtlb-miss	%tim	dtsb-miss	%tim	%tim
0	u	8k	30	0.0	0	0.0	2170236	46.1	0	0.0	46.1
0	u	64k	0	0.0	0	0.0	0	0.0	0	0.0	0.0
0	u	512k	0	0.0	0	0.0	0	0.0	0	0.0	0.0
0	u	4m	0	0.0	0	0.0	0	0.0	0	0.0	0.0
0	k	8k	1	0.0	0	0.0	4174	0.1	10	0.0	0.1
0	k	64k	0	0.0	0	0.0	0	0.0	0	0.0	0.0
0	k	512k	0	0.0	0	0.0	0	0.0	0	0.0	0.0
0	k	4m	0	0.0	0	0.0	0	0.0	0	0.0	0.0
ttl			31	0.0	0	0.0	2174410	46.2	10	0.0	46.2

When to use large pages

High TLB miss

```
so19# cpustat -c pic0=Cycle_cnt,pic1=DTLB_miss 1
time  cpu  event  pic0      pic1
1.006  0    tick  663839993  3540016
2.006  0    tick  651843834  3514443
3.006  0    tick  630482518  3398061
4.006  0    tick  651910256  3511458
5.006  0    tick  651432039  3510201
6.006  0    tick  663839993  3309406
7.006  0    tick  650806115  3510292
```

Programming MME

- A wrapper program, `ppgsz(1)`
- A preload library, `libmpss.so.1`
- Compiling options
- A programmatic interface, `memcntl(2)`

A Wrapper program, ppgsize

```

sol9# pmap -sx `pgrep testprog`
2909:  ./testprog
  Address  Kbytes      RSS      Anon  Locked  Pgsz  Mode   Mapped File
00010000      8         8        -     -     8K  r-x--  dev:277,83 ino:114875
00020000      8         8         8     -     8K  rwx--  dev:277,83 ino:114875
00022000 131088    131088 131088     -     8K  rwx--  [ heap ]
FF280000     120        120        -     -     8K  r-x--  libc.so.1
FF29E000     136        128        -     -     -   r-x--  libc.so.1
FF2C0000      72         72        -     -     8K  r-x--  libc.so.1
FF2D2000     192        192        -     -     -   r-x--  libc.so.1
FF302000     112        112        -     -     8K  r-x--  libc.so.1
FF31E000      48         32        -     -     -   r-x--  libc.so.1
FF33A000      24         24        24     -     8K  rwx--  libc.so.1
FF340000      8          8         8     -     8K  rwx--  libc.so.1
FF390000      8          8         -     -     8K  r-x--  libc_psr.so.1
FF3A0000      8          8         -     -     8K  r-x--  libdl.so.1
FF3B0000      8          8         8     -     8K  rwx--  [ anon ]
FF3C0000     152        152        -     -     8K  r-x--  ld.so.1
FF3F6000      8          8         8     -     8K  rwx--  ld.so.1
FFBFA000      24         24        24     -     8K  rwx--  [ stack ]
-----
total Kb 132024 132000 131168     -

```

A Wrapper program, ppgsize

```

sol9# ppgsz -o heap=4M ./testprog &
sol9# pmap -sx `pgrep testprog`
2953:  ./testprog
  Address  Kbytes      RSS      Anon  Locked  Pgsz  Mode   Mapped File
00010000      8         8        -     -     8K  r-x--  dev:277,83 ino:114875
00020000      8         8         8     -     8K  rwx--  dev:277,83 ino:114875
00022000    3960    3960    3960     -     8K  rwx--  [ heap ]
00400000  131072  131072  131072     -     4M  rwx--  [ heap ]
FF280000     120     120        -     -     8K  r-x--  libc.so.1
FF29E000     136     128        -     -     -  r-x--  libc.so.1
FF2C0000      72      72        -     -     8K  r-x--  libc.so.1
FF2D2000     192     192        -     -     -  r-x--  libc.so.1
FF302000     112     112        -     -     8K  r-x--  libc.so.1
FF31E000      48      32        -     -     -  r-x--  libc.so.1
FF33A000      24      24       24     -     8K  rwx--  libc.so.1
FF340000      8         8         8     -     8K  rwx--  libc.so.1
FF390000      8         8        -     -     8K  r-x--  libc_psr.so.1
FF3A0000      8         8        -     -     8K  r-x--  libdl.so.1
FF3B0000      8         8         8     -     8K  rwx--  [ anon ]
FF3C0000     152     152        -     -     8K  r-x--  ld.so.1
FF3F6000      8         8         8     -     8K  rwx--  ld.so.1
FFBFA000      24      24       24     -     8K  rwx--  [ stack ]
-----
total Kb  135968  135944  135112     -

```

A Preload library, libmpss.so.1

```
MPSSHEAP=size
MPSSSTACK=size
MPSSHEAP and MPSSSTACK specify the preferred page
sizes for the heap and stack, respectively. The speci-
fied page size(s) are applied to all created
processes.
size must be a supported page size (see pagesize(1))
or 0, in which case the system will select an
appropriate page size (see memcntl(2)).
size can be qualified with K, M, G, or T to specify
Kilobytes, Megabytes, Gigabytes, or Terabytes respec-
tively.
MPSSCFGFILE=config-file
config-file is a text file which contains one or more
mpss configuration entries of the form:
exec-name exec-args:heap-size:stack-size
```

```
example$ LD_PRELOAD=$LD_PRELOAD:mpss.so.1
example$ MPSSHEAP=512K
example$ MPSSSTACK=64K
example$ export LD_PRELOAD MPSSHEAP MPSSSTACK
```

```
example$ LD_PRELOAD=$LD_PRELOAD:mpss.so.1
example$ MPSSCFGFILE=mpsscfg
example$ export LD_PRELOAD MPSSCFGFILE
example$ cat $MPSSCFGFILE
foo*:512K:64K
```

Compiling Options

SunForte 8 compiler

- -xpagesize=n
- -xpagesize_heap=n
- -xpagesize_stack=n

n value

- 8K, 64K, 512k, 4M, 32M, 256M,
2G, 16G

A programmatic interface, memcntl

```
#include <sys/types.h>
#include <sys/mman.h>
#include <stdlib.h>

#define MEGABYTE ((size_t) (1024 * 1024))
#define FOUR_MEGABYTE ((size_t) 4 * MEGABYTE)

int main(int ac, char *av[])
{
    struct memcntl_mha mha;
    char *my_memory;

    mha.mha_cmd = MHA_MAPSIZE_BSSBRK;
    mha.mha_flags = 0;
    mha.mha_pagesize = FOUR_MEGABYTE;
    memcntl(NULL, 0, MC_HAT_ADVISE, (char *) &mha, 0, 0);
    my_memory = (char *) memalign(FOUR_MEGABYTE, (size_t) 100 * MEGABYTE);
}
```

Discovering Supported Page Sizes

- `sr1-cse108-08:/home/jaehyuk % pagesize`
8192
`sr1-cse108-08:/home/jaehyuk % pagesize -a`
8192
65536
524288
4194304
- `meminfo()` - 2
- `getpagesize()` - 3c
- `getpagesizes()` - 3c

Commands & API Quick-Reference

ppgsz	An administrative wrapper program for advising page size preferences. ppgsz is not inherited across exec() by a new program.
pmap -sx	A utility to print the MMU page size for each mapping in the program.
mpss.so.1	A preload library, enabled by setting LD_PRELOAD=mpss.so.1 for advising page size preferences for existing applications. Advise is held across exec() of a new program.
trapstat -t	A tool for measuring the amount of time spent servicing TLB misses.
cc	New options are included in the SunOne Studio 8 compiler -xpagesize_heap and -xpagesize_stack

Thank You!

Jaehyuk.Lee@sun.com