On Providing Useful Information for Analyzing and Tuning Applications

John Mellor−Crummey and Robert Fowler

Computer Science Dept., Rice University

David Whalley

Computer Science Dept., Florida State University

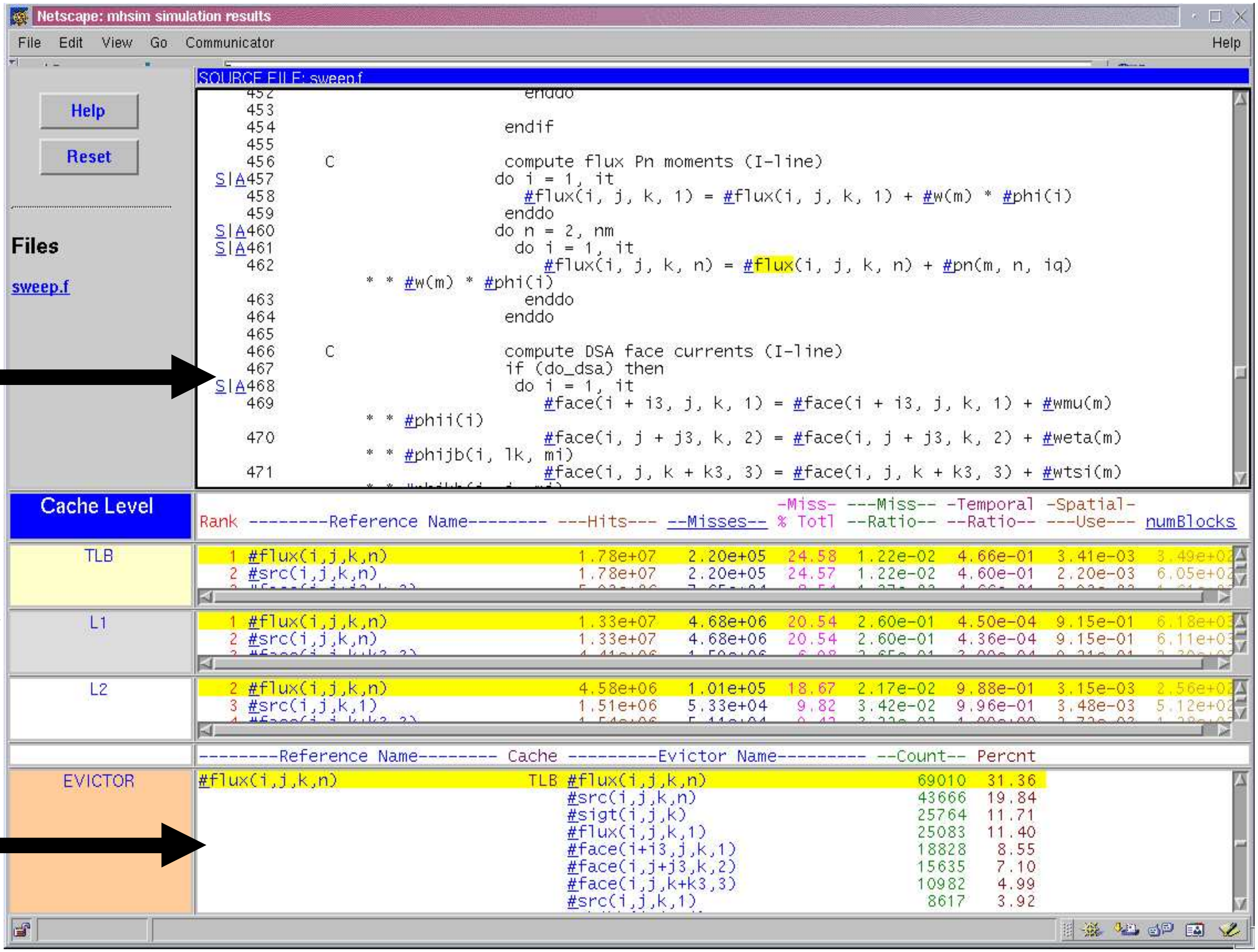
Our Approach for Understanding Application Performance

- Gather performance data from multiple sources.
  - static analysis, profilers, simulation
- Fuse data, compute derived metrics, and correlate with the source code into a hyperlinked database.
- View database using commodity hypertext browsing technology.
- Tools
  - MHSIM: simulator for multi−level memory hierarchies
  - HPCView: correlates multiple profiles

MHSIM

- Used for diagnosing memory hierarchy performance bottlenecks.
- F77 source code is instrumented with calls to a memory hierarchy simulator.
- Provides performance information for each level of the memory hierarchy for source references, loops, and arrays.
- Provides measures for locality, spatial use, and bytes accessed.
- Indicates for each reference the set of references that cause it to be evicted.

The MHSIM User Interface


Netscape: mhsim simulation results
File Edit View Go Communicator
Help
Help
Reset
Files
sweep.f
Cache Level
TLB
L1
L2
EVICTOR

SOURCE FILE: sweep.f

```
                                  enddo

                                endif

          C                     compute flux Pn moments (I-line)
S|A                            do i = 1, it
                                  #flux(i, j, k, 1) = #flux(i, j, k, 1) + #w(m) * #phi(i)
                                enddo
S|A                            do n = 2, nm
S|A                              do i = 1, it
                                    #flux(i, j, k, n) = #flux(i, j, k, n) + #pn(m, n, iq)
                  * * #w(m) * #phi(i)
                                  enddo
                                enddo

          C                     compute DSA face currents (I-line)
                                if (do_dsa) then
S|A                              do i = 1, it
                                    #face(i + i3, j, k, 1) = #face(i + i3, j, k, 1) + #wmu(m)
                  * * #phii(i)
                                    #face(i, j + j3, k, 2) = #face(i, j + j3, k, 2) + #weta(m)
                  * * #phijb(i, lk, mi)
                                    #face(i, j, k + k3, 3) = #face(i, j, k + k3, 3) + #wtsi(m)
```

```
                                                    -Miss-  ---Miss-- -Temporal -Spatial-
Rank --------Reference Name-------- ---Hits--- --Misses-- % Totl --Ratio-- --Ratio-- ---Use--- numBlocks
TLB
   1 #flux(i,j,k,n)                  1.78e+07   2.20e+05  24.58  1.22e-02  4.66e-01  3.41e-03  3.49e+02
   2 #src(i,j,k,n)                   1.78e+07   2.20e+05  24.57  1.22e-02  4.60e-01  2.20e-03  6.05e+02
L1
   1 #flux(i,j,k,n)                  1.33e+07   4.68e+06  20.54  2.60e-01  4.50e-04  9.15e-01  6.18e+03
   2 #src(i,j,k,n)                   1.33e+07   4.68e+06  20.54  2.60e-01  4.36e-04  9.15e-01  6.11e+03
L2
   2 #flux(i,j,k,n)                  4.58e+06   1.01e+05  18.67  2.17e-02  9.88e-01  3.15e-03  2.56e+02
   3 #src(i,j,k,1)                   1.51e+06   5.33e+04   9.82  3.42e-02  9.96e-01  3.48e-03  5.12e+02
```

```
--------Reference Name-------- Cache ---------Evictor Name--------- --Count-- Percnt
#flux(i,j,k,n)                  TLB #flux(i,j,k,n)                     69010   31.36
                                    #src(i,j,k,n)                      43666   19.84
                                    #sigt(i,j,k)                       25764   11.71
                                    #flux(i,j,k,1)                     25083   11.40
                                    #face(i+i3,j,k,1)                  18828    8.55
                                    #face(i,j+j3,k,2)                  15635    7.10
                                    #face(i,j,k+k3,3)                  10982    4.99
                                    #src(i,j,k,1)                       8617    3.92
```

Source Code

Cache Info

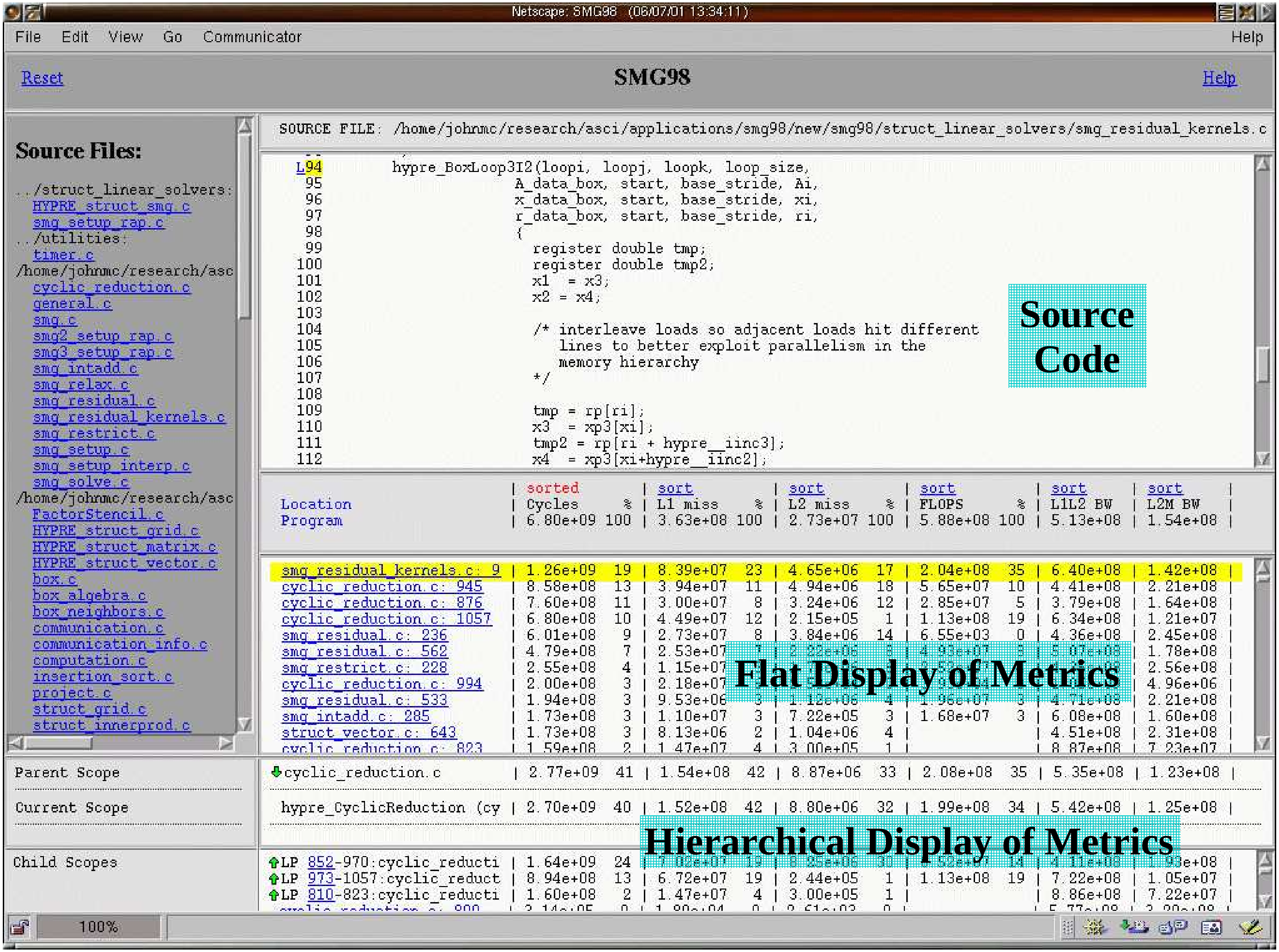
Eviction Info

Selection in any pane navigates other panes and highlights matching information.

HPCView

- Reads one or more performance metric measurement files stored in machine−independent XML form (e.g. metrics gathered from hardware performance counter sampling).
- A companion binary analyzer supplies loop nesting information to enable HPCView to compute loop−level performance metrics.
- Can compute derived performance metrics as functions of input metrics.
- Correlates performance metrics with source code producing structured output at line, loop, procedure, file, and program levels.
- Completely customizable views: input metrics, program structure information, and computed metric equations all from a configuration file.

The HPCView User Interface



Netscape: SMG98 (06/07/01 13:34:11)
File Edit View Go Communicator Help
Reset
SMG98
Help
100%

Source Files:

../struct_linear_solvers:
- HYPRE_struct_smg.c
- smg_setup_rap.c

../utilities:
- timer.c

/home/johnmc/research/asc
- cyclic_reduction.c
- general.c
- smg.c
- smg2_setup_rap.c
- smg3_setup_rap.c
- smg_intadd.c
- smg_relax.c
- smg_residual.c
- smg_residual_kernels.c
- smg_restrict.c
- smg_setup.c
- smg_setup_interp.c
- smg_solve.c

/home/johnmc/research/asc
- FactorStencil.c
- HYPRE_struct_grid.c
- HYPRE_struct_matrix.c
- HYPRE_struct_vector.c
- box.c
- box_algebra.c
- box_neighbors.c
- communication.c
- communication_info.c
- computation.c
- insertion_sort.c
- project.c
- struct_grid.c
- struct_innerprod.c

SOURCE FILE: /home/johnmc/research/asci/applications/smg98/new/smg98/struct_linear_solvers/smg_residual_kernels.c

```
      hypre_BoxLoop3I2(loopi, loopj, loopk, loop_size,
                       A_data_box, start, base_stride, Ai,
                       x_data_box, start, base_stride, xi,
                       r_data_box, start, base_stride, ri,
                       {
                         register double tmp;
                         register double tmp2;
                         x1  = x3;
                         x2 = x4;

                         /* interleave loads so adjacent loads hit different
                            lines to better exploit parallelism in the
                            memory hierarchy
                         */

                         tmp = rp[ri];
                         x3  = xp3[xi];
                         tmp2 = rp[ri + hypre__iinc3];
                         x4  = xp3[xi+hypre__iinc2];
```

Source Code

```
                          | sorted         | sort           | sort           | sort           | sort       | sort       |
 Location                 | Cycles       % | L1 miss      % | L2 miss      % | FLOPS        % | L1L2 BW    | L2M BW     |
 Program                  | 6.80e+09 100   | 3.63e+08 100   | 2.73e+07 100   | 5.88e+08 100   | 5.13e+08   | 1.54e+08   |

 smg_residual_kernels.c: 9 | 1.26e+09  19  | 8.39e+07  23   | 4.65e+06  17   | 2.04e+08  35   | 6.40e+08   | 1.42e+08   |
 cyclic_reduction.c: 945  | 8.58e+08  13   | 3.94e+07  11   | 4.94e+06  18   | 5.65e+07  10   | 4.41e+08   | 2.21e+08   |
 cyclic_reduction.c: 876  | 7.60e+08  11   | 3.00e+07   8   | 3.24e+06  12   | 2.85e+07   5   | 3.79e+08   | 1.64e+08   |
 cyclic_reduction.c: 1057 | 6.80e+08  10   | 4.49e+07  12   | 2.15e+05   1   | 1.13e+08  19   | 6.34e+08   | 1.21e+07   |
 smg_residual.c: 236      | 6.01e+08   9   | 2.73e+07   8   | 3.84e+06  14   | 6.55e+03   0   | 4.36e+08   | 2.45e+08   |
 smg_residual.c: 562      | 4.79e+08   7   | 2.53e+07   7   | 2.22e+06   8   | 4.93e+07   3   | 5.07e+08   | 1.78e+08   |
 smg_restrict.c: 228      | 2.55e+08   4   | 1.15e+07 [illegible] | [illegible] | [illegible] | [illegible] | 2.56e+08   |
 cyclic_reduction.c: 994  | 2.00e+08   3   | 2.18e+07 [illegible] | [illegible] | [illegible] | [illegible] | 4.96e+06   |
 smg_residual.c: 533      | 1.94e+08   3   | 9.53e+06   3   | 1.12e+06   4   | 1.96e+07   3   | 4.71e+08   | 2.21e+08   |
 smg_intadd.c: 285        | 1.73e+08   3   | 1.10e+07   3   | 7.22e+05   3   | 1.68e+07   3   | 6.08e+08   | 1.60e+08   |
 struct_vector.c: 643     | 1.73e+08   3   | 8.13e+06   2   | 1.04e+06   4   |                | 4.51e+08   | 2.31e+08   |
 cyclic_reduction.c: 823  | 1.59e+08   2   | 1.47e+07   4   | 3.00e+05   1   |                | 8.87e+08   | 7.23e+07   |
```

Flat Display of Metrics

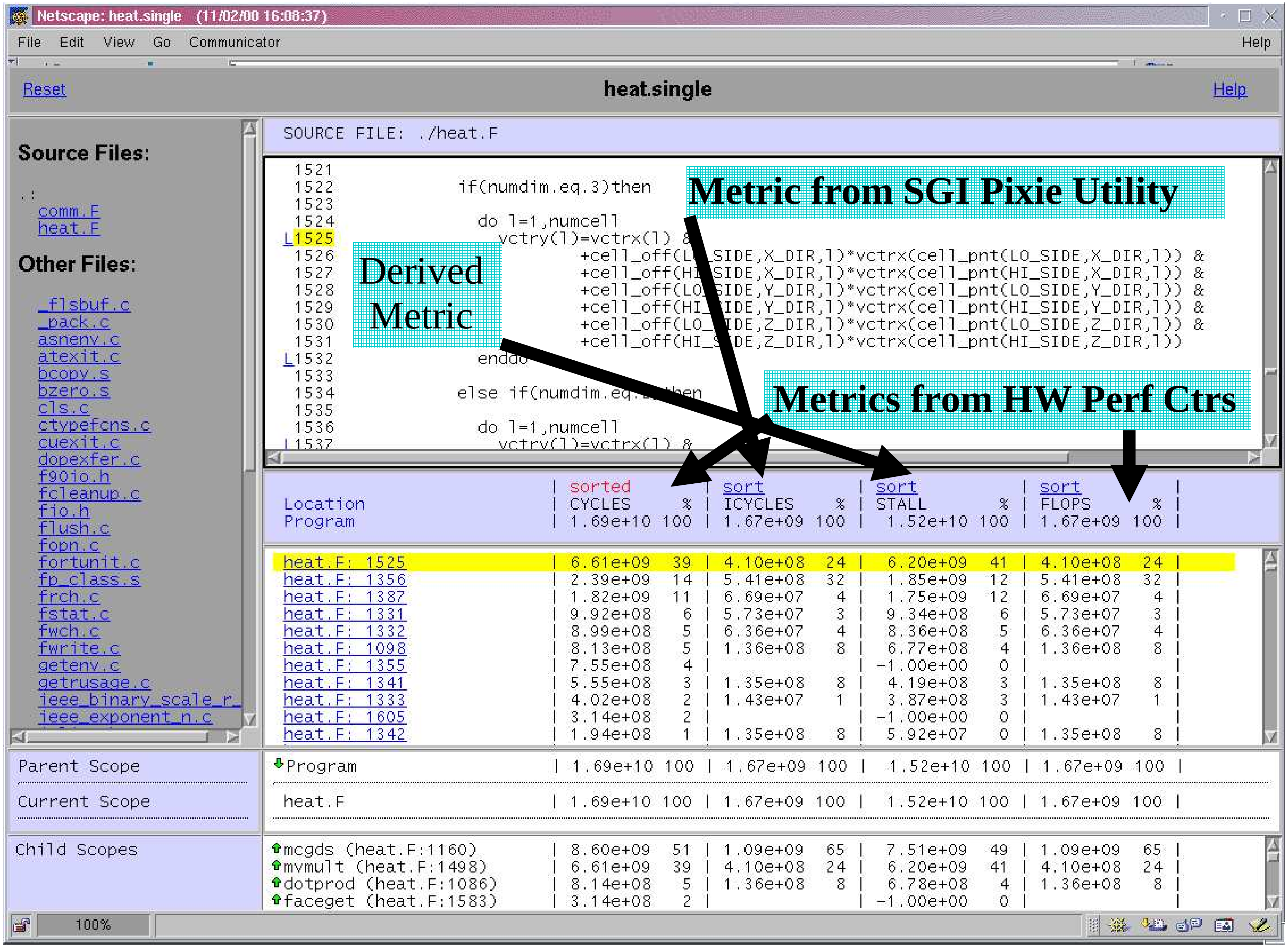
```
Parent Scope     ⇩cyclic_reduction.c          | 2.77e+09  41 | 1.54e+08  42 | 8.87e+06  33 | 2.08e+08  35 | 5.35e+08 | 1.23e+08 |

Current Scope     hypre_CyclicReduction (cy   | 2.70e+09  40 | 1.52e+08  42 | 8.80e+06  32 | 1.99e+08  34 | 5.42e+08 | 1.25e+08 |

Child Scopes     ⇧LP 852-970:cyclic_reducti   | 1.64e+09  24 | 7.02e+07  19 | 8.25e+06  30 | 8.52e+07  14 | 4.11e+08 | 1.93e+08 |
                 ⇧LP 973-1057:cyclic_reduct   | 8.94e+08  13 | 6.72e+07  19 | 2.44e+05   1 | 1.13e+08  19 | 7.22e+08 | 1.05e+07 |
                 ⇧LP 810-823:cyclic_reducti   | 1.60e+08   2 | 1.47e+07   4 | 3.00e+05   1 |              | 8.86e+08 | 7.22e+07 |
                  cyclic_reduction.c: 800     | 2.14e+05   0 | 1.80e+04   0 | 2.61e+03   0 |              | 5.77e+08 | 3.20e+08 |
```

Hierarchical Display of Metrics

HPCView Displaying a Derived Metric



heat.single

Source Files:

./
comm.F
heat.F

Other Files:

_flsbuf.c
_pack.c
asnenv.c
atexit.c
bcopy.s
bzero.s
cls.c
ctypefcns.c
cuexit.c
dopexfer.c
f90io.h
fcleanup.c
fio.h
flush.c
fopn.c
fortunit.c
fp_class.s
frch.c
fstat.c
fwch.c
fwrite.c
getenv.c
getrusage.c
ieee_binary_scale_r_
ieee_exponent_n.c

SOURCE FILE: ./heat.F

```
 1521
 1522           if(numdim.eq.3)then
 1523
 1524             do l=1,numcell
L1525               vctry(l)=vctrx(l) &
 1526                   +cell_off(LO_SIDE,X_DIR,l)*vctrx(cell_pnt(LO_SIDE,X_DIR,l)) &
 1527                   +cell_off(HI_SIDE,X_DIR,l)*vctrx(cell_pnt(HI_SIDE,X_DIR,l)) &
 1528                   +cell_off(LO_SIDE,Y_DIR,l)*vctrx(cell_pnt(LO_SIDE,Y_DIR,l)) &
 1529                   +cell_off(HI_SIDE,Y_DIR,l)*vctrx(cell_pnt(HI_SIDE,Y_DIR,l)) &
 1530                   +cell_off(LO_SIDE,Z_DIR,l)*vctrx(cell_pnt(LO_SIDE,Z_DIR,l)) &
 1531                   +cell_off(HI_SIDE,Z_DIR,l)*vctrx(cell_pnt(HI_SIDE,Z_DIR,l))
L1532             enddo
 1533
 1534           else if(numdim.eq.2)then
 1535
 1536             do l=1,numcell
L1537               vctry(l)=vctrx(l) &
```

Metric from SGI Pixie Utility

Derived Metric

Metrics from HW Perf Ctrs

```
Location                | sorted              | sort              | sort               | sort              |
Program                 | CYCLES          %   | ICYCLES       %   | STALL          %   | FLOPS         %   |
                        | 1.69e+10  100       | 1.67e+09  100     |  1.52e+10  100     | 1.67e+09  100     |

heat.F: 1525            | 6.61e+09   39       | 4.10e+08   24     |  6.20e+09   41     | 4.10e+08   24     |
heat.F: 1356            | 2.39e+09   14       | 5.41e+08   32     |  1.85e+09   12     | 5.41e+08   32     |
heat.F: 1387            | 1.82e+09   11       | 6.69e+07    4     |  1.75e+09   12     | 6.69e+07    4     |
heat.F: 1331            | 9.92e+08    6       | 5.73e+07    3     |  9.34e+08    6     | 5.73e+07    3     |
heat.F: 1332            | 8.99e+08    5       | 6.36e+07    4     |  8.36e+08    5     | 6.36e+07    4     |
heat.F: 1098            | 8.13e+08    5       | 1.36e+08    8     |  6.77e+08    4     | 1.36e+08    8     |
heat.F: 1355            | 7.55e+08    4       |                   | -1.00e+00    0     |                   |
heat.F: 1341            | 5.55e+08    3       | 1.35e+08    8     |  4.19e+08    3     | 1.35e+08    8     |
heat.F: 1333            | 4.02e+08    2       | 1.43e+07    1     |  3.87e+08    3     | 1.43e+07    1     |
heat.F: 1605            | 3.14e+08    2       |                   | -1.00e+00    0     |                   |
heat.F: 1342            | 1.94e+08    1       | 1.35e+08    8     |  5.92e+07    0     | 1.35e+08    8     |

Parent Scope   Program            | 1.69e+10  100 | 1.67e+09  100 |  1.52e+10  100 | 1.67e+09  100 |
Current Scope  heat.F             | 1.69e+10  100 | 1.67e+09  100 |  1.52e+10  100 | 1.67e+09  100 |

Child Scopes   mcgds (heat.F:1160)    | 8.60e+09   51 | 1.09e+09   65 |  7.51e+09   49 | 1.09e+09   65 |
               mvmult (heat.F:1498)   | 6.61e+09   39 | 4.10e+08   24 |  6.20e+09   41 | 4.10e+08   24 |
               dotprod (heat.F:1086)  | 8.14e+08    5 | 1.36e+08    8 |  6.78e+08    4 | 1.36e+08    8 |
               faceget (heat.F:1583)  | 3.14e+08    2 |               | -1.00e+00    0 |               |
```

Summary

- MHSIM provides detailed memory hierarchy utilization information via simulation.
- HPCView displays multiple sources of performance information and computes derived metrics.
- Results are shown in scope−hierarchy views of the source code and the output is produced as HTML databases that can be displayed using commodity browsers.
- Has been used to improve the performance of large scientific applications.