CrayPat-lite

Heidi Poxon
Manager & Technical Lead, Performance Tools
Cray Inc.

CrayPat-lite Goals

- **Provide automatic application performance statistics at the end of a job**
  - Focus is to offer a simplified interface to basic application performance information for users not familiar with the Cray performance tools and perhaps new to application performance analysis
  - Provides a simple performance summary mechanism for Cray performance tools users before they move on to more detailed analysis with classic perftools
  - Gives sites the option to enable/disable application performance data collection for all users for a period of time

- **Keep traditional or "classic" perftools working the same as before**

- **Provide a simple way to transition from perftools-lite to perftools to encourage further tool use for performance analysis**

Steps to Using CrayPat “classic”

Access performance tools software

```
> module load perftools
```

Build program, retaining .o files

```
> make
```

→ a.out

Instrument binary

```
> pat_build –O apa a.out
```

→ a.out+pat

Modify batch script and run program

```
aprun a.out+pat
```

→ a.out+pat*.xf

Process raw performance data and create report

```
> pat_report a.out+pat*.xf
```

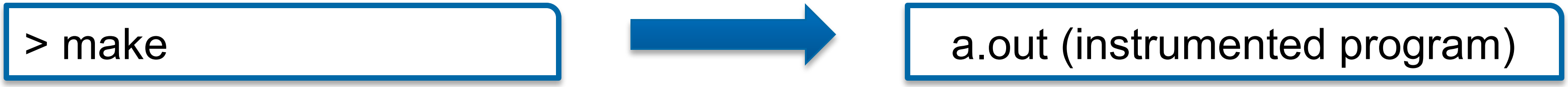
→ a.out+pat*.ap2
Text report to stdout
a.out+pat*.apa
MPICH_RANK_XXX

Steps to Using CrayPat-lite

Access light version of performance tools software

```
> module load perftools-lite
```

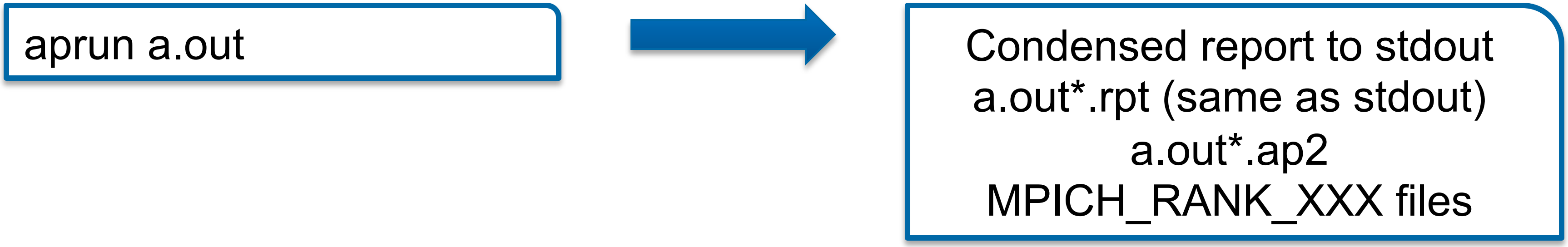
Build program



```
> make
```

a.out (instrumented program)

Run program (no modification to batch script)



```
aprun a.out
```

Condensed report to stdout
a.out*.rpt (same as stdout)
a.out*.ap2
MPICH_RANK_XXX files

Benefits of CrayPat-lite

- Program is automatically relinked to add instrumentation in a.out (pat_build step done for the user)
- .o files are automatically preserved
- No modifications are needed to a batch script to run instrumented binary, since original binary is replaced with instrumented version
- pat_report is automatically run before job exits
- Performance statistics are issued to stdout
- User can use “classic” CrayPat for more in-depth performance investigation

Predefined Set of Performance Experiments

- **Set of predefined experiments, enabled with the CRAYPAT_LITE environment variable**
  - sample_profile
  - event_profile
  - GPU

What do the predefined events mean to someone familiar with the Cray performance tools?

CRAYPAT_LITE=sample_profile

- **Default experiment**
- **Equivalent to "`pat_build –O apa a.out`"**
- **Provides profile based on sampling**
  - Includes collection of summary CPU performance counters around MAIN (for MFLOPS)
  - Includes Imbalance information
- **More information available in .ap2 file**
  - Can get classic report by running pat_report

CRAYPAT_LITE=event_profile

- **Provides profile based on summarization of events**

- **Includes OpenMP and OpenACC information if these models are used within program**

- **Equivalent to "`pat_build –u –gmpi a.out`" +**
  - Collection of summary CPU performance counters
  - Filter to only trace functions above 1200 bytes
    - In most cases, omits tiny repetitive functions that can perturb results (like ranf())
    - Can give coarser granularity results over classic perftools

- **More information available in .ap2 file**

CRAYPAT_LITE=GPU

- **Provides more detailed OpenACC GPU statistics**
- **Equivalent to "`pat_build –w a.out`" (coarsest granularity tracing, around MAIN)**
- **Output similar to classic perftools accelerator table**
  - Includes host and device time
  - Bytes transferred between host and device
  - Time to transfer data between host and device
- **More information available in .ap2 file**

Performance Statistics Available

- **Set of predefined experiments, enabled with the CRAYPAT_LITE environment variable**
  - Sample_profile
  - Event_profile
  - GPU

- **Job information**
  - Number of MPI ranks, ranks per node, number of threads
  - Wallclock
  - Memory high water mark
  - Aggregate MFLOPS (CPU only)

- **Profile of top time consuming routines with load balance**
- **Observations**
- **Instructions on how to get more information**

Sample CrayPat-lite Report (sample_profile)

```
#################################################################
#                                                               #
#            CrayPat-lite Performance Statistics                #
#                                                               #
#################################################################

CrayPat/X:  Version 6.1.0.10929 Revision 10929 (xf 10658)  03/04/13 23:51:00
Experiment:                  lite  sample_profile
Number of PEs (MPI ranks):     64
Numbers of PEs per Node:       32  PEs on each of  2  Nodes
Numbers of Threads per PE:      1
Number of Cores per Socket:    16
Execution start time:  Tue Mar  5 16:40:56 2013
System name and speed:  mork 2100 MHz

Wall Clock Time:    145.228474 secs
High Memory:            201.00 MBytes
MFLOPS (aggregate):   35559.45 M/sec

Table 1:  Profile by Function Group and Function (top 10 functions shown)

 Samp%  |    Samp  | Imb.  |  Imb.  |Group
        |          | Samp  | Samp%  | Function
        |          |       |        |  PE=HIDE

 100.0% | 14272.5 |    -- |     -- |Total
|----------------------------------------------------
|  46.0% |  6561.4 |    -- |     -- |USER
||---------------------------------------------------
||    5.9% |   847.6 | 155.4 |  15.7% |collocate_core_1_
||    4.9% |   700.3 | 125.7 |  15.5% |integrate_core_2_
||    3.8% |   544.0 | 124.0 |  18.9% |collocate_core_2_
||    3.7% |   523.1 |  73.9 |  12.6% |integrate_core_1_
||===================================================
|  29.7% |  4239.6 |    -- |     -- |MPI
||---------------------------------------------------
||    9.3% |  1328.3 | 198.7 |  13.2% |mpi_alltoallv
||    4.2% |   598.5 |  71.5 |  10.8% |mpi_waitall
||    2.9% |   413.8 | 107.2 |  20.9% |MPI_WAITANY
||    2.9% |   409.1 |  66.9 |  14.3% |MPI_Comm_create
||===================================================
…
```

Sample CrayPat-lite Report (For More Info…)

```
Program invocation:
  cp2k.x H2O-64.inp

For more detailed performance reports, run:
  pat_report /lus/scratch/cp2k.x.ap2

For interactive performance analysis, run:
  app2 /lus/scratch/cp2k.x.ap2

End of CrayPat output.
```

Sample CrayPat-lite Report (event_profile)

```
#################################################################
#                                                               #
#            CrayPat-lite Performance Statistics                #
#                                                               #
#################################################################

CrayPat/X:  Version 6.1.0.10863 Revision 10863 (xf 10658)  02/13/13 15:23:08
Experiment:                  lite  event_profile
Number of PEs (MPI ranks):   64
Numbers of PEs per Node:     32  PEs on each of  2  Nodes
Numbers of Threads per PE:    1
Number of Cores per Socket:  16
Execution start time:  Fri Feb 15 14:42:24 2013

Wall Clock Time:  122.608994 secs
High Memory:  45.70 MBytes
MFLOPS (aggregate):  15763.16 M/sec


Table 1:  Profile by Function Group and Function (top 7 functions shown)

 Time%   |      Time  |     Imb. |   Imb. |     Calls   |Group
         |            |     Time |  Time% |             | Function
         |            |          |        |             |  PE=HIDE

 100.0% | 101.961423 |       -- |     -- | 5315211.9 |Total
|-----------------------------------------------------------------------------------
|  92.5% |  94.267451 |       -- |     -- | 5272245.9 |USER
||----------------------------------------------------------------------------------
||  75.8% |  77.248585 | 2.356249 |   3.0% |    1001.0 |LAMMPS_NS::PairLJCut::compute
||   6.5% |   6.644545 | 0.105246 |   1.6% |      51.0 |LAMMPS_NS::Neighbor::half_bin_newton
||   4.1% |   4.131842 | 0.634032 |  13.5% |       1.0 |LAMMPS_NS::Verlet::run
||   3.8% |   3.841349 | 1.241434 |  24.8% | 5262868.9 |LAMMPS_NS::Pair::ev_tally
||   1.3% |   1.288463 | 0.181268 |  12.5% |    1000.0 |LAMMPS_NS::FixNVE::final_integrate
||==================================================================================
|   7.0% |   7.110931 |       -- |     -- |   42637.0 |MPI
||----------------------------------------------------------------------------------
||   4.8% |   4.851309 | 3.371093 |  41.6% |   12267.0 |MPI_Send
||   1.5% |   1.536106 | 2.592504 |  63.8% |   12267.0 |MPI_Wait
|===================================================================================
```

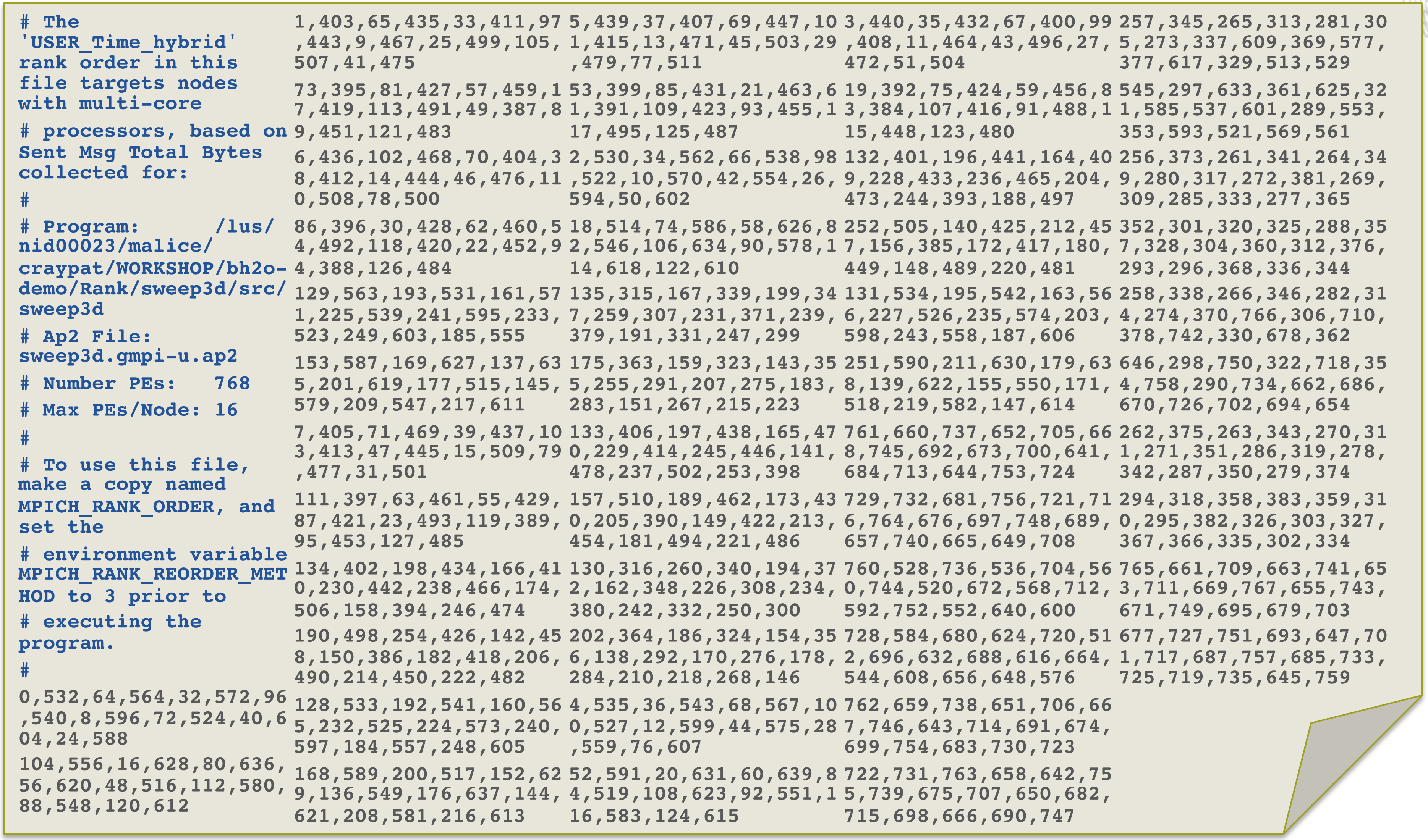
Sample CrayPat-lite Report (Observations)

```
================  Observations and suggestions  ========================

MPI Grid Detection:

There appears to be point-to-point MPI communication in a 4 X 2 X 8 grid
    pattern. The execution time spent in MPI functions might be reduced
    with a rank order that maximizes communication between ranks on the
    same node. The effect of several rank orders is estimated below.

    A file named MPICH_RANK_ORDER.Grid was generated along with this
    report and contains usage instructions and the Hilbert rank order
    from the following table.

         Rank    On-Node     On-Node   MPICH_RANK_REORDER_METHOD
        Order   Bytes/PE   Bytes/PE%
                            of Total
                            Bytes/PE

      Hilbert  5.533e+10      90.66%   3
         Fold  4.907e+10      80.42%   2
          SMP  4.883e+10      80.02%   1
   RoundRobin  3.740e+10      61.28%   0
```

MPICH_RANK_ORDER File

```
# The 'Custom' rank order in this file targets nodes with multi-core
# processors, based on Sent Msg Total Bytes collected for:
#
# Program:     /lus/nid00030/heidi/sweep3d/mod/sweep3d.mpi
# Ap2 File:    sweep3d.mpi+pat+27054-89t.ap2
# Number PEs:  48
# Max PEs/Node: 4
#
# To use this file, make a copy named MPICH_RANK_ORDER, and set the
# environment variable MPICH_RANK_REORDER_METHOD to 3 prior to
# executing the program.
#
# The following table lists rank order alternatives and the grid_order
# command-line options that can be used to generate a new order.
…
```

Auto-Generated MPI Rank Order File



```
# The 'USER_Time_hybrid' rank order in this file targets nodes with multi-core
# processors, based on Sent Msg Total Bytes collected for:
#
# Program:      /lus/nid00023/malice/craypat/WORKSHOP/bh2o-demo/Rank/sweep3d/src/sweep3d
# Ap2 File: sweep3d.gmpi-u.ap2
# Number PEs:   768
# Max PEs/Node: 16
#
# To use this file, make a copy named MPICH_RANK_ORDER, and set the
# environment variable MPICH_RANK_REORDER_METHOD to 3 prior to
# executing the program.
#
0,532,64,564,32,572,96,540,8,596,72,524,40,604,24,588
104,556,16,628,80,636,56,620,48,516,112,580,88,548,120,612
1,403,65,435,33,411,97,443,9,467,25,499,105,507,41,475
73,395,81,427,57,459,17,419,113,491,49,387,89,451,121,483
6,436,102,468,70,404,38,412,14,444,46,476,110,508,78,500
86,396,30,428,62,460,54,492,118,420,22,452,94,388,126,484
129,563,193,531,161,571,225,539,241,595,233,523,249,603,185,555
153,587,169,627,137,635,201,619,177,515,145,579,209,547,217,611
7,405,71,469,39,437,103,413,47,445,15,509,79,477,31,501
111,397,63,461,55,429,87,421,23,493,119,389,95,453,127,485
134,402,198,434,166,410,230,442,238,466,174,506,158,394,246,474
190,498,254,426,142,458,150,386,182,418,206,490,214,450,222,482
128,533,192,541,160,565,232,525,224,573,240,597,184,557,248,605
168,589,200,517,152,629,136,549,176,637,144,621,208,581,216,613
5,439,37,407,69,447,101,415,13,471,45,503,29,479,77,511
53,399,85,431,21,463,61,391,109,423,93,455,117,495,125,487
2,530,34,562,66,538,98,522,10,570,42,554,26,594,50,602
18,514,74,586,58,626,82,546,106,634,90,578,114,618,122,610
135,315,167,339,199,347,259,307,231,371,239,379,191,331,247,299
175,363,159,323,143,355,255,291,207,275,183,283,151,267,215,223
133,406,197,438,165,470,229,414,245,446,141,478,237,502,253,398
157,510,189,462,173,430,205,390,149,422,213,454,181,494,221,486
130,316,260,340,194,372,162,348,226,308,234,380,242,332,250,300
202,364,186,324,154,356,138,292,170,276,178,284,210,218,268,146
4,535,36,543,68,567,100,527,12,599,44,575,28,559,76,607
52,591,20,631,60,639,84,519,108,623,92,551,116,583,124,615
3,440,35,432,67,400,99,408,11,464,43,496,27,472,51,504
19,392,75,424,59,456,83,384,107,416,91,488,115,448,123,480
132,401,196,441,164,409,228,433,236,465,204,473,244,393,188,497
252,505,140,425,212,457,156,385,172,417,180,449,148,489,220,481
131,534,195,542,163,566,227,526,235,574,203,598,243,558,187,606
251,590,211,630,179,638,139,622,155,550,171,518,219,582,147,614
761,660,737,652,705,668,745,692,673,700,641,684,713,644,753,724
729,732,681,756,721,716,764,676,697,748,689,657,740,665,649,708
760,528,736,536,704,560,744,520,672,568,712,592,752,552,640,600
728,584,680,624,720,512,696,632,688,616,664,544,608,656,648,576
762,659,738,651,706,667,746,643,714,691,674,699,754,683,730,723
722,731,763,658,642,755,739,675,707,650,682,715,698,666,690,747
257,345,265,313,281,305,273,337,609,369,577,377,617,329,513,529
545,297,633,361,625,321,585,537,601,289,553,353,593,521,569,561
256,373,261,341,264,349,280,317,272,381,269,309,285,333,277,365
352,301,320,325,288,357,328,304,360,312,376,293,296,368,336,344
258,338,266,346,282,314,274,370,766,306,710,378,742,330,678,362
646,298,750,322,718,354,758,290,734,662,686,670,726,702,694,654
262,375,263,343,270,311,271,351,286,319,278,342,287,350,279,374
294,318,358,383,359,310,295,382,326,303,327,367,366,335,302,334
765,661,709,663,741,653,711,669,767,655,743,671,749,695,679,703
677,727,751,693,647,701,717,687,757,685,733,725,719,735,645,759
```

Sample Report (CRAYPAT_LITE=GPU)

```
###################################################################
#                                                                 #
#            CrayPat-lite Performance Statistics                  #
#                                                                 #
###################################################################

CrayPat/X:  Version 6.1.0.10929 Revision 10929 (xf 10658)  03/04/13 23:51:00
Experiment:                   lite  gpu
Number of PEs (MPI ranks):       8
Numbers of PEs per Node:         1  PE on each of   8  Nodes
Numbers of Threads per PE:       1
Number of Cores per Socket:     16
Execution start time:  Tue Mar  5 17:50:34 2013
System name and speed:  mork 2100 MHz

Table 1:  Time and Bytes Transferred for Accelerator Regions

  Host  |  Host  |   Acc  |  Acc Copy  | Acc Copy  | Events  |Calltree
 Time%  |  Time  |  Time  |        In  |      Out  |         | PE=HIDE
        |        |        |  (MBytes)  | (MBytes)  |         |

 100.0% | 25.572 | 24.201 |       2555 |       2560 |   38164 |Total
|-------------------------------------------------------------------------------------------------------
| 100.0% | 25.572 | 24.201 |       2555 |       2560 |   38164 |himenobmtxp_
|        |        |        |            |            |         | himenobmtxp_.ACC_DATA_REGION@li.65
3 100.0% | 25.568 | 24.144 |       2555 |       2560 |   38152 |  jacobi_
4        |        |        |            |            |         |   jacobi_.ACC_DATA_REGION@li.227
|||||---------------------------------------------------------------------------------------------------
5||||  73.5% | 18.792 | 18.507 |      0.004 |      0.004 |    5015 |jacobi_.ACC_REGION@li.309
6||||  72.2% | 18.467 |     -- |         -- |         -- |    1003 | jacobi_.ACC_SYNC_WAIT@li.331
5||||  15.2% |  3.878 |  1.130 |         -- |       2555 |    3009 |jacobi_.ACC_UPDATE@li.382
||||||--------------------------------------------------------------------------------------------------
6|||||  10.2% |  2.613 |     -- |         -- |         -- |    1003 |jacobi_.ACC_SYNC_WAIT@li.382
6|||||   4.9% |  1.262 |  1.130 |         -- |       2555 |    1003 |jacobi_.ACC_COPY@li.382
||||||==================================================================================================
5||||   6.9% |  1.763 |  1.629 |       2555 |         -- |    2006 |jacobi_.ACC_UPDATE@li.271
5||||   6.9% |  1.761 |  1.629 |       2555 |         -- |    1003 | jacobi_.ACC_COPY@li.271
5||||   3.4% |  0.857 |     -- |         -- |         -- |       2 |jacobi_.ACC_DATA_REGION@li.227(exclusive)
|=======================================================================================================
```

Questions ?