BERKELEY LAB

NERSC

NATIONAL ENERGY RESEARCH
SCIENTIFIC COMPUTING CENTER

Franklin Profiling and Performance Tools

Jonathan Carter

NUG October 2008 Meeting

Office of Science
U.S. DEPARTMENT OF ENERGY

Overview

- Why use profiling and performance tools?
- What information can tools provide?
- Tools available
  - IPM
  - Craypat
  - Apprentice2
- Recommendations

Why use profiling tools?

- Understand and improve application performance
  - Computation and communication
- Determine application concurrency sweet spot
- ERCAP

Considerations

- How much time will profiling experiments take?
  - Use appropriate data volumes and concurrencies
  - Tradeoff of science vs. tuning
- Interpreting data
  - Some data is easy to interpret, e.g. MPI_Barrier() time
  - Other data requires a performance model to understand whether a measurement is good or bad
- Tool Robustness
  - Increased data volume and concurrency increases chance of tool failure

Information from Tools

- Events
  - Routine called
  - Hardware counters sampled
  - Memory high-water mark changes
  - Message sent
- Traces
  - Temporal record of events
- Profiles
  - Inventory of events

Opteron Hardware Performance Counters

- Opteron has 4 hardware performance counters
  - Native Opteron hardware events, targeted to hardware engineers
- Most tools use PAPI to provide an abstraction to map these native events
  - PAPI_FP_OPS is number of flops
- Cray groups these counters into sets of four from which derived metrics can be presented
  - Group 3 is bandwidth metrics
  - man hwpc to see others

IPM

- Integrated Performance Monitoring
  - Originally developed at NERSC, now Open Source NSF-funded project
  - Compact report: memory, flop/s, communication vs. computation breakdown
  - For Franklin, need to link in IPM

```
module load ipm
export IPM_REPORT=terse
#export IPM_REPORT=full
ftn -o a.out b.f $IPM
aprun -n 64 a.out
```

IPM

```
##IPMv0.921##########################################################
#
# command : qlg8  (completed)
# host    : nid03562/x86_64_Linux          mpi_tasks : 64 on 16 nodes
# start   : 09/29/08/22:16:29              wallclock : 62.206852 sec
# stop    : 09/29/08/22:17:31              %comm     : 1.21
# gbytes  : 1.29125e+01 total              gflop/sec : 4.19667e+01 tot
#
#####################################################################
```

IPM

```
#                          [total]         <avg>           min           max
# entries                       64             1             1             1
# wallclock                3981.13       62.2052       62.2043       62.2069
# user                     3954.69        61.792       61.6799       61.8799
# system                   35.0462      0.547596      0.456028      0.660041
# mpi                      48.3425      0.755351      0.464974      0.929935
# %comm                                  1.21426      0.747482       1.49491
# gflop/sec                41.9667      0.655729      0.655229      0.656079
# gbytes                   12.9125      0.201759      0.201756      0.201763
#
# PAPI_FP_OPS            2.61062e+12   4.07909e+10   4.07597e+10   4.08126e+10
# PAPI_TOT_CYC           9.06027e+12   1.41567e+11   1.40972e+11   1.41831e+11
# PAPI_VEC_INS           4.71447e+12   7.36636e+10   7.36131e+10   7.36943e+10
# PAPI_TOT_INS           1.03574e+13   1.61835e+11   1.60745e+11   1.62557e+11
#
#                           [time]       [calls]        <%mpi>      <%wall>
# MPI_Wait                 36.7927        161280         76.11         0.92
# MPI_Irecv                7.88529         80640         16.31         0.20
# MPI_Isend                3.63493         80640          7.52         0.09
# MPI_Reduce             0.0276335           384          0.06         0.00
# MPI_Gather            0.00178182            64          0.00         0.00
# MPI_Comm_size         9.90945e-05            64          0.00         0.00
# MPI_Comm_rank         3.06483e-05            64          0.00         0.00
```

Office of Science
U.S. DEPARTMENT OF ENERGY

Craypat

- Tracing and profiling framework provided by Cray
  - Produce an instrumented version of your application that can be used for tracing and profiling experiments
  - Run instrumented application to produce profile or trace output
  - Run reporting tools to analyze data

pat_hwpc

- **Build instrumented executable, run, and report in a single shot**

```
module load xt-craypat
ftn -o a.out b.f
pat_hwpc aprun -n 64 a.out
```

pat_hwpc

Imb = Max - Avg

Four counters

Time%		100.0%	
Time		62.645081	
Imb.Time		0.427314	
Imb.Time%		0.7%	
Calls		404	
PAPI_L1_DCM	1.328M/sec	82543475	misses
PAPI_TOT_INS	2649.722M/sec	164668587582	instr
PAPI_L1_DCA	1315.445M/sec	81749173603	refs
PAPI_FP_OPS	647.455M/sec	40236472777	ops
User time (approx)	62.146 secs	142934923125	cycles
Average Time per Call sec/call		0.155062	
Overhead / Time		0.0%	
Cycles	62.146 secs	142934923125	cycles
User time (approx)	62.146 secs	142934923125	cycles

pat_hwpc

```
Utilization rate                                  99.2%
Instr per cycle                                    1.15 inst/cycle
HW FP Ops / Cycles                                 0.28 ops/cycle
HW FP Ops / User time      647.455M/sec   40236472777 ops
 7.0%peak(DP)
HW FP Ops / WCT            642.293M/sec
HW FP Ops / Inst                                  24.4%
Computation intensity                              0.49 ops/ref
MIPS                     169582.18M/sec
MFLOPS (aggregate)        41437.10M/sec
Instructions per LD ST                             2.01 inst/ref
LD & ST per D1 miss                              990.38 refs/miss
D1 cache hit ratio                                99.9%
LD ST per Instructions                            49.6%
```

Per core

All cores

Derived L1 DC metrics

pat_build

- **Build an instrumented executable**

```
module load xt-craypat
ftn -o a.out b.f
pat_build a.out
aprun -n 64 a.out+pat
```

- **Trace user, library, or system functions**
  - `-u trace` all user functions in application
  - `-g mpi` trace all MPI functions
  - `-g io` trace all IO functions

```
module load xt-craypat
ftn -o a.out b.f
pat_build -g mpi -u a.out
aprun -n 64 a.out+pat
```

pat_build

- Exclude functions individually or in lists

```
module load xt-craypat
ftn -o a.out b.f
pat_build -g mpi -T !MPI_Bcast -t list1 a.out
aprun -n 64 a.out+pat
```

- Trace all MPI except `MPI_Bcast` and trace all functions listed in file `list1`

Pat API

```
PAT_region_begin(id, label, istat)
PAT_region_end(id, istat)
PAT_record(cmd, istat)
PAT_profiling_state(cmd, istat)
PAT_sampling_state(cmd, istat)
PAT_tracing_state(cmd, istat)

id: region identifier
label: region label
cmd: on, off, query
istat: return code
```

Running an instrumented application

- Experiments (PAT_RT_EXPERIMENT)
  - `trace`
    - Default when tracing user, library, or system functions
    - Can collect hardware counters for each traced function
  - `samp_pc_prof`
    - Default when no functions traced
    - Lowest overhead, samples program counter, no call stack, or performance counters
  - `samp_pc_time`
    - Samples program counter and time, can collect performance counters

Running an instrumented application (2)

- Collecting hardware counters (PAT_RT_HPWC)
  - 0 Summary with instruction metrics
  - 1 Summary with TLB metrics
  - 2 L1 and L2 metrics
  - 3 Bandwidth information
  - 4 HT information (*not supported for QC*)
  - 5 Floating point mix
  - 6 Cycles stalled, resources idle
  - 7 Cycles stalled, resources full
  - 8 Instructions and branches
  - 9 Instruction cache

Running an instrumented application (3)

- Controlling trace file size
  - PAT_RT_CALLSTACK=1 limit call stack data to parent, default is trace back to top level
  - PAT_RT_SUMMARY=0 turn off runtime summarization. *Leads to bigger trace file, but required for Apprentice2*

- Summary

```
export PAT_RT_EXPERIMENT=trace
export PAT_RT_HWPC=2
export PAT_RT_CALLSTACK=1
aprun -n 64 a.out+pat
```

pat_report

- pat_report aggregates the profile or trace data into one or more reports
- Prints date and time of run, executable and environment
- Reasonable defaults for most experiments, can add reports with -O *option*
  - mpi
  - load_balance
  - program_time
- Create unique reports with -d *data* data extraction and -b *option* aggregation options

pat_report

- samp_pc_prof experiment
- Sample counts by function
- ETC contains system routines

Aggregated by

```
Table 1:  Profile by Function

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

 100.0% | 3896 |    -- |     -- |Total
|-----------------------------------------
|  91.5% | 3566 |    -- |     -- |USER
||----------------------------------------
||  88.8% | 3461 | 23.41 |   0.7% |stream_collide_
||   0.7% |   29 |  8.41 |  23.1% |check_
||   0.5% |   19 |  8.02 |  30.2% |stream1m_
||   0.5% |   18 |  8.53 |  32.1% |stream1p_
||   0.4% |   14 |  5.03 |  26.9% |MAIN_
||   0.2% |    8 |  7.25 |  49.1% |stream2m_
||   0.2% |    8 |  5.86 |  42.5% |stream2p_
||   0.1% |    5 |  5.17 |  52.5% |stream3p_
||   0.1% |    4 |  4.86 |  54.9% |stream3m_
||========================================
|   8.5% |  330 |    -- |     -- |ETC
```

Functions with most samples

pat_report

- samp_pc_prof experiment
- Sample counts by line

```
Table 2:  Profile by Group, Function, and Line

 Samp % | Samp |  Imb. |   Imb. |Group
        |      |  Samp | Samp % | Function
        |      |       |        |  Source
        |      |       |        |   Line
        |      |       |        |    PE='HIDE'

 100.0% | 3886 |    -- |     -- |Total
|-----------------------------------------------
|  91.7% | 3562 |    -- |     -- |USER
||----------------------------------------------
||  88.9% | 3456 |    -- |     -- |stream_collide_
3|        |      |       |        |qlg.v8.f
||||--------------------------------------------
4|||   2.8% |  108 | 16.95 |  13.8% |line.627
4|||   2.9% |  111 |  3.47 |   3.1% |line.635
4|||   3.0% |  116 |  3.44 |   2.9% |line.644
4|||   3.3% |  130 |  3.67 |   2.8% |line.653
```

pat_report

```
USER / stream_collide_
----------------------------------------------------------------------
  Samp%                                          88.8%
  Samp                                            3469
  Imb.Samp                                       28.02
  Imb.Samp%                                       0.8%
  PAPI_L1_DCM                1.316M/sec       72554164 misses
  PAPI_TLB_DM                0.281M/sec       15512146 misses
  PAPI_L1_DCA             1333.683M/sec    73545960344 refs
  PAPI_FP_OPS              659.392M/sec    36362182403 ops
  User time (approx)        55.145 secs   126833500000 cycles
  Cycles                    55.145 secs   126833500000 cycles
  User time (approx)        55.145 secs   126833500000 cycles
  HW FP Ops / Cycles                               0.29 ops/cycle
  HW FP Ops / User time    659.392M/sec    36362182403 ops        7.2%peak(DP)
  Computation intensity                            0.49 ops/ref
  MFLOPS (aggregate)      42201.10M/sec
  LD & ST per TLB miss                          4741.19 refs/miss
  LD & ST per D1 miss                           1013.67 refs/miss
  D1 cache hit ratio                              99.9%
  % TLB misses / cycle                             0.0%
```

Four counters

Derived metrics

- samp_pc_time experiment, with HWPC group 0
- HW counts by function, line

Office of Science
U.S. DEPARTMENT OF ENERGY

pat_report

- trace expt
- MPI and user functions traced

Table 1: Profile by Function Group and Function

Time %	Time	Imb. Time	Imb. Time %	Calls	Group Function PE='HIDE'
100.0%	62.585786	--	--	6389	Total
98.0%	61.315812	--	--	1324	USER
93.2%	58.299623	0.381214	0.7%	51	stream_collide_
1.8%	1.108034	0.023878	2.1%	1	MAIN_
1.4%	0.855313	0.001645	0.2%	6	check_
0.5%	0.306938	0.031895	9.6%	210	stream1p_
0.5%	0.305643	0.025736	7.9%	210	stream1m_
0.2%	0.134554	0.024962	15.9%	210	stream2m_
0.2%	0.130652	0.019315	13.1%	210	stream2p_
0.1%	0.092032	0.004700	4.9%	210	stream3p_
0.1%	0.082845	0.003903	4.6%	210	stream3m_
2.0%	1.267441	--	--	5058	MPI
1.1%	0.670412	0.247204	27.4%	2520	mpi_wait_
0.7%	0.411681	0.385606	49.1%	1	mpi_cart_create_
0.2%	0.118653	0.019418	14.3%	1260	mpi_irecv_
0.1%	0.066546	0.009511	12.7%	1260	mpi_isend_

Derived from times not samples

pat_report

Max, Med, Min

- trace expt
- MPI and user functions traced

```
Table 2:  Load Balance with MPI Sent Message Stats

 Time % |      Time |  Sent |  Sent Msg |  Avg Sent |Group
        |           |   Msg |     Total |  Msg Size | PE[mmm]
        |           | Count |     Bytes |           |

 100.0% | 62.589245 |  1260 | 201600000 | 160000.00 |Total
|-----------------------------------------------------------
|  98.0% | 61.316529 |    -- |        -- |        -- |USER
||----------------------------------------------------------
||   1.5% | 61.672472 |    -- |        -- |        -- |pe.36
||   1.5% | 61.303518 |    -- |        -- |        -- |pe.45
||   1.5% | 61.080109 |    -- |        -- |        -- |pe.21
||==========================================================
|   2.0% |  1.270179 |  1260 | 201600000 | 160000.00 |MPI
||----------------------------------------------------------
||   0.0% |  1.852124 |  1260 | 201600000 | 160000.00 |pe.14
||   0.0% |  1.223305 |  1260 | 201600000 | 160000.00 |pe.30
||   0.0% |  0.505516 |  1260 | 201600000 | 160000.00 |pe.36
```

pat_report

- trace expt
- MPI and user functions traced
- Where are messages sent from?

```
Table 3:  MPI Sent Message Stats by Caller

  Sent Msg |  Sent | 64KB<= |Function
     Total |   Msg |  MsgSz | Caller
     Bytes | Count |   <1MB |  PE[mmm]
           |       |  Count |

 201600000 |  1260 |   1260 |Total
|-------------------------------------
| 201600000 |  1260 |   1260 |mpi_isend_
||------------------------------------
||  33600000 |   210 |    210 |stream3p_
|||-----------------------------------
3||  32640000 |   204 |    204 |stream_collide_
4||           |       |        | MAIN_
|||||---------------------------------
5||||  32640000 |   204 |    204 |pe.33
5||||  32640000 |   204 |    204 |pe.54
5||||  32640000 |   204 |    204 |pe.5
|||||=================================
3||    960000 |     6 |      6 |check_
4||           |       |        | MAIN_
|||||---------------------------------
5||||    960000 |     6 |      6 |pe.33
5||||    960000 |     6 |      6 |pe.54
5||||    960000 |     6 |      6 |pe.5
```

pat_report

```
USER / stream_collide_
----------------------------------------------------------------------
  Time%                                          93.1%
  Time                                       58.151747
  Imb.Time                                    0.194460
  Imb.Time%                                       0.3%
  Calls                                             51
  PAPI_L1_DCM                1.088M/sec      63306976 misses
  PAPI_TLB_DM                0.223M/sec      12964023 misses
  PAPI_L1_DCA             1343.009M/sec   78121062876 refs
  PAPI_FP_OPS              678.827M/sec   39486486432 ops
  User time (approx)        58.169 secs  133787941719 cycles
  Average Time per Call                      0.000000 sec/call
  Overhead / Time                           275806.5%
  Cycles                    58.169 secs  133787941719 cycles
  User time (approx)        58.169 secs  133787941719 cycles
  Utilization rate                             100.0%
  HW FP Ops / Cycles                             0.30 ops/cycle
  HW FP Ops / User time    678.827M/sec   39486486432 ops        7.4%peak(DP)
  HW FP Ops / WCT          678.827M/sec
  Computation intensity                          0.51 ops/ref
  MFLOPS (aggregate)      43444.95M/sec
  LD & ST per TLB miss                        6025.99 refs/miss
  LD & ST per D1 miss                         1234.00 refs/miss
  D1 cache hit ratio                            99.9%
  % TLB misses / cycle                           0.0%
```

- trace experiment, with HWPC group 0
- HW counts by function

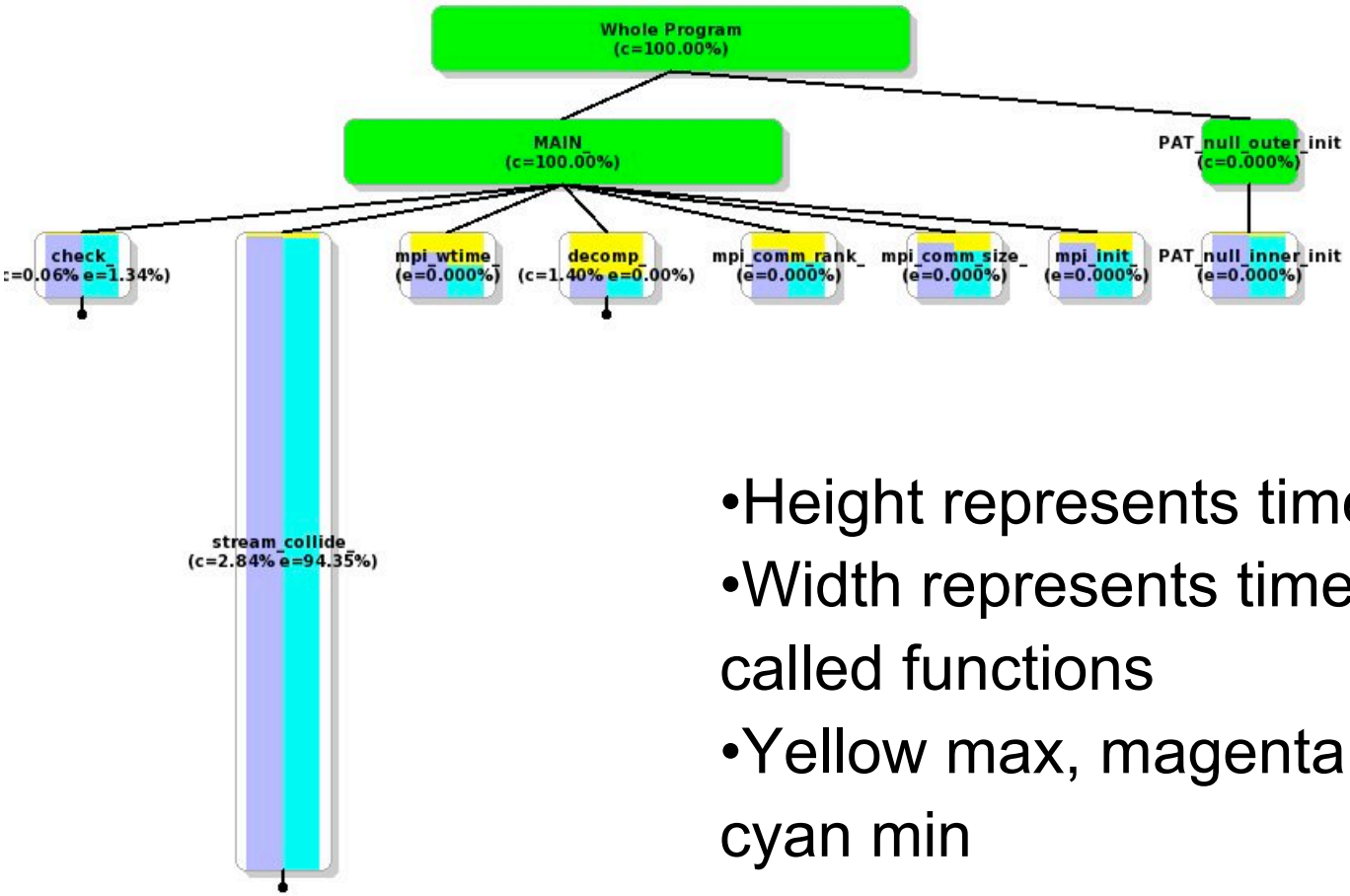
Derived from times not samples

Apprentice2

- Tool to visualize
  - Time, msg, hardware counter profiles as charts
  - Call tree
  - Message volume between pairs of tasks (need PAT_RT_SUMMARY=0)
  - MPI timeline (need PAT_RT_SUMMARY=0)

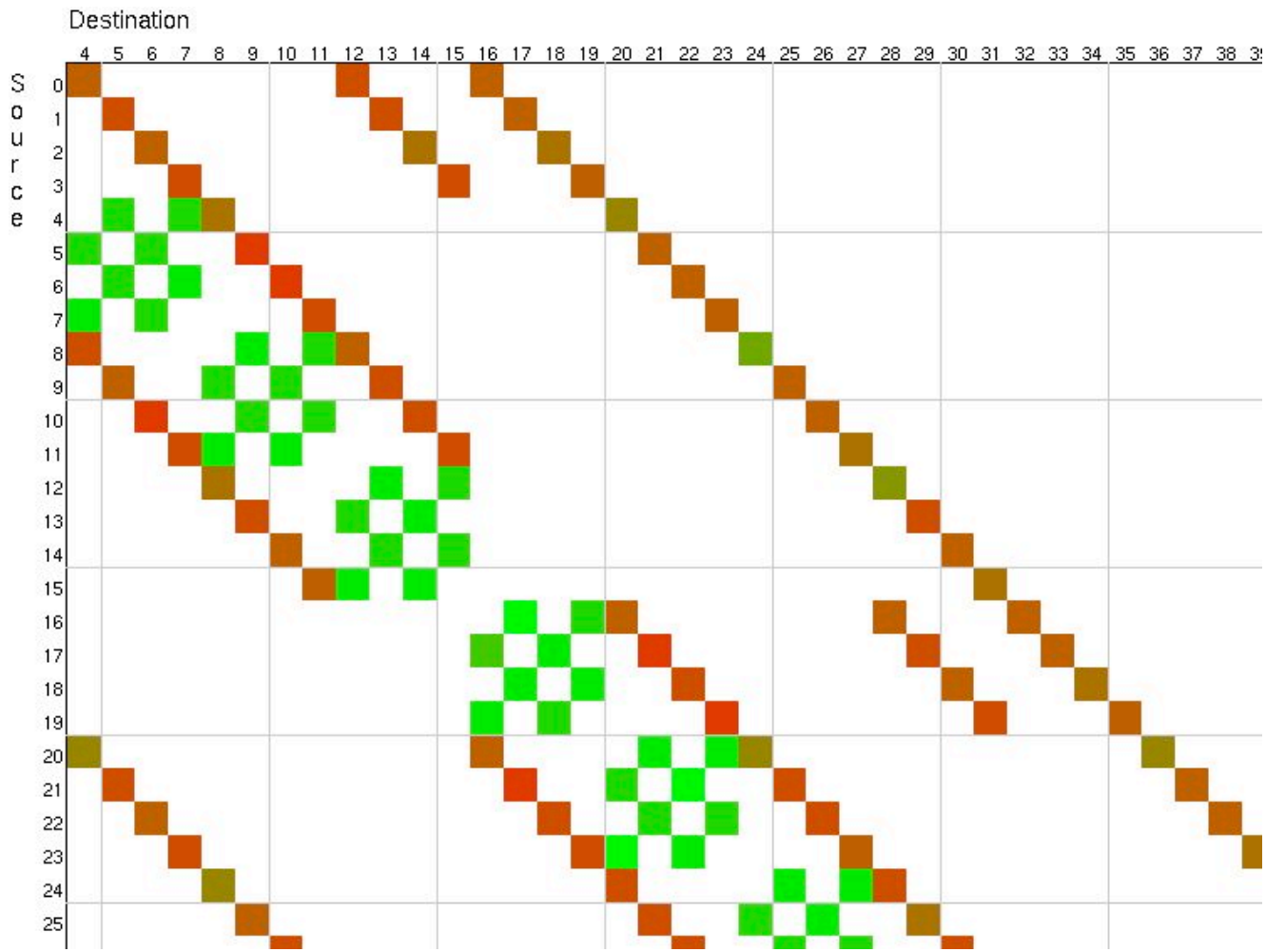
```
pat_report qlg8+pat+18200-1472tdt.xf
app2 qlg8+pat+18200-1472tdt.ap2
```

Apprentice2 Call Tree

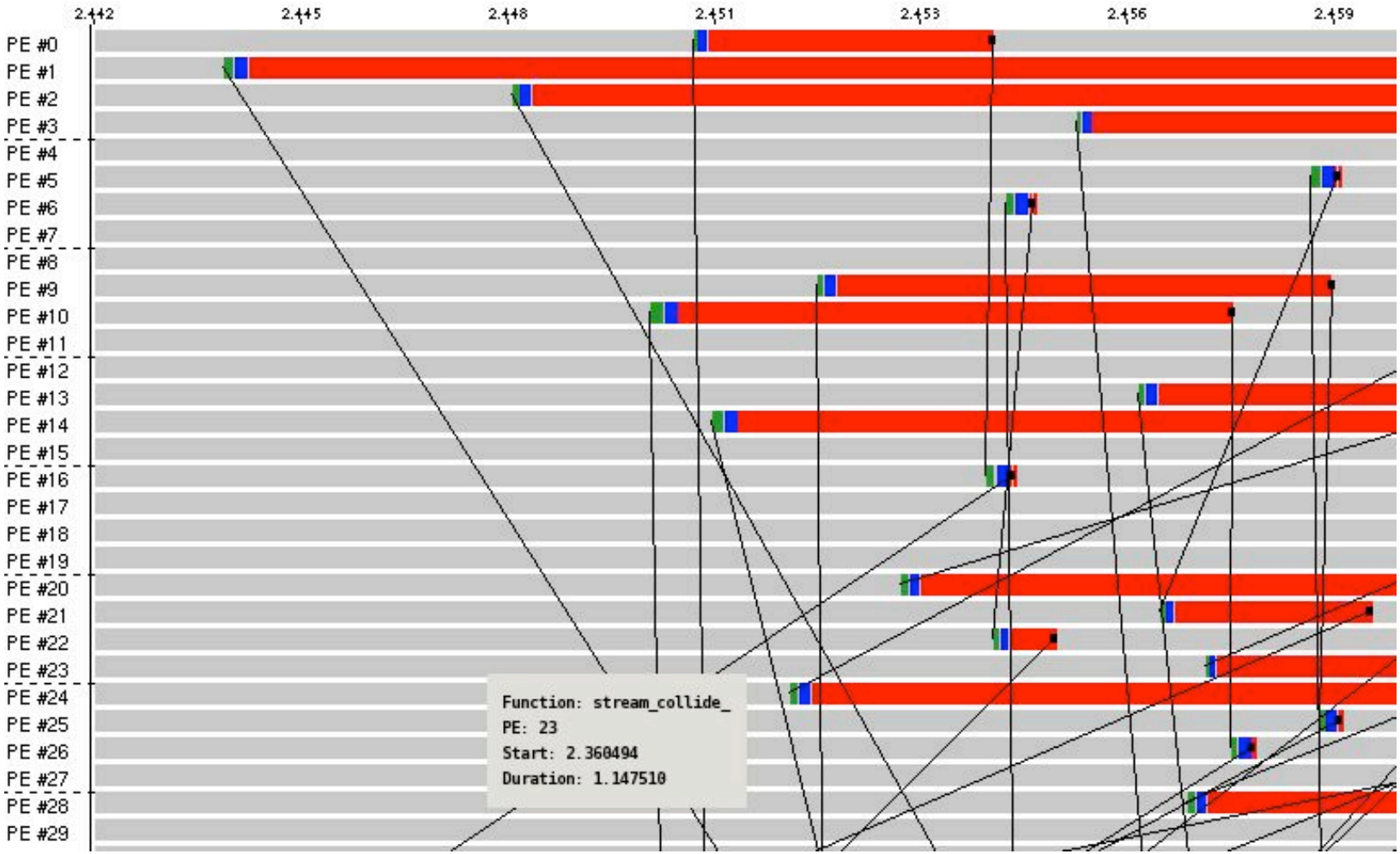


- Height represents time
- Width represents time in called functions
- Yellow max, magenta avg, cyan min

Apprentice2 Msg/Pairs



Apprentice2 Timeline



Recommendations

- Easiest to use
  - IPM for snapshot of performance
  - pat_hwpc (non-default) hardware performance counters for whole application
  - pat_build, run with sampling experiment and pat_report for profile of application hotspots (no options required)

Recommendations

- Performance of individual functions
  - pat_build, run with tracing experiment and pat_report for profile of functions
  - Consider API to limit file size
  - If tracing is too expensive, try time sampling with large time increment

Further Information

- Cray Documents
  - Using Cray Performance Analysis Tools S-2376-41
    - Examples are for dual-core Opteron, ignore Chapter 4.
  - man pages: intro_craypat, pat_build, pat_report, hwpc, apprentice, pat_help
- NUG 2007 talks on Optimizing For XT4
- AMD Documents
  - BIOS and Kernel Developer's Guide For AMD Family 10h Processors
  - Software Optimization Guide for AMD Family 10h Processors