TACC
TEXAS ADVANCED COMPUTING CENTER
WWW.TACC.UTEXAS.EDU

TEXAS
The University of Texas at Austin

Tools for Monitoring CPU Usage and Affinity in Multicore Supercomputers

SC19
Denver, CO | hpc is now.

HUST-19

Nov 18, 2019

PRESENTED BY:

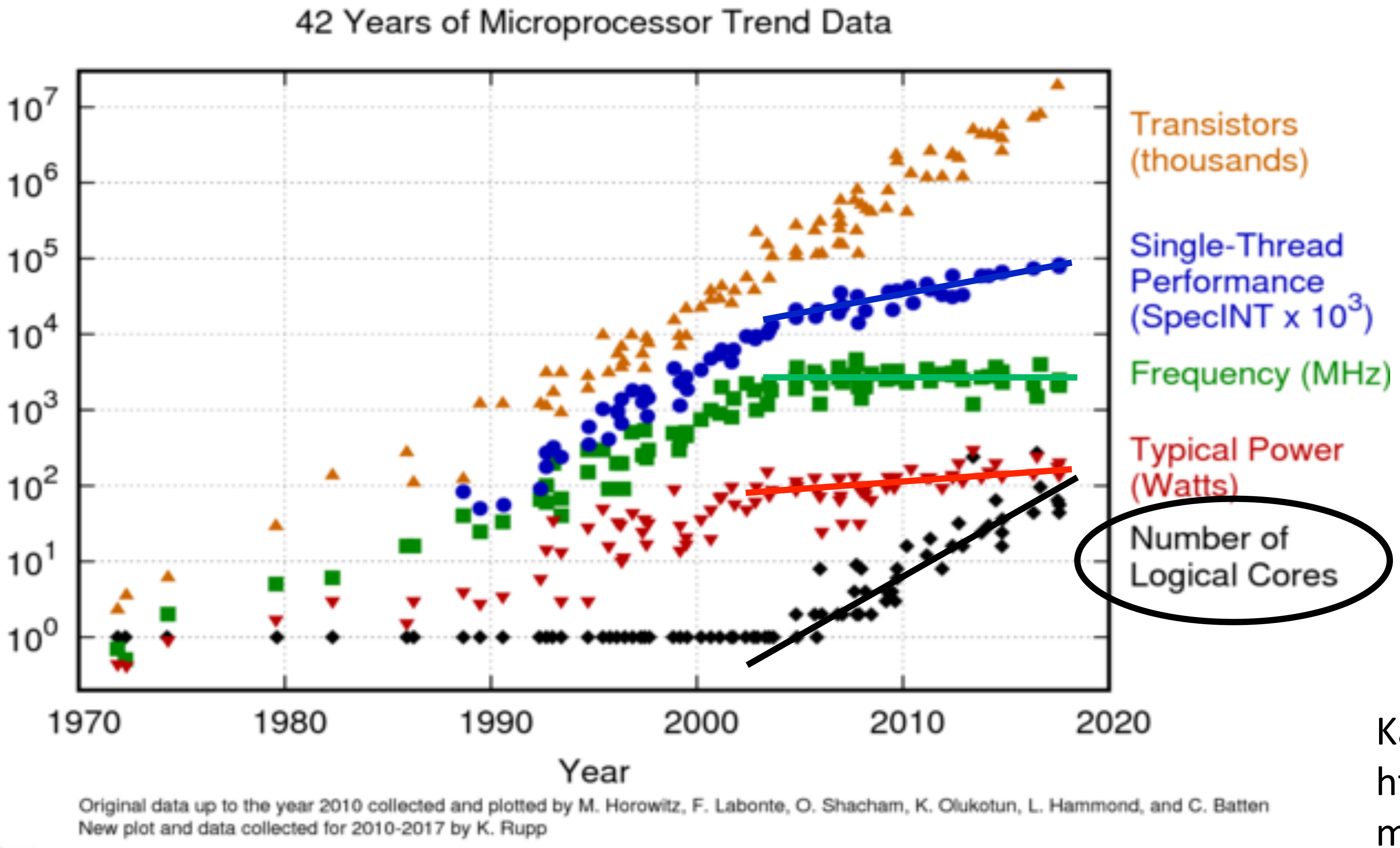
Kent Milfeld

Lei Huang

Si Liu

(milfeld|huang|siliu@tacc.utexas.edu)

Microprocessor Trend Data (1970-2020)



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

Karl Rupp
https://github.com/karlrupp/microprocessor-trend-data

Motivation

- HPC Nodes: many cores, sockets, SMT/HyperThreading
  - NUMA Nodes, PCIe location, Tiles, etc. -- **it's complicated**
  - Many Tools for System Managers (”not users”)
  - HPC User approach to running a job:
    - Finally got the app to compile, or thank goodness for site-installed apps!
    - `sbatch job` where job contains `mpirun app`

- Sometimes you need to peek inside a node!

Innovative Tools

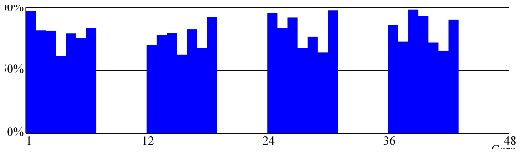
- Innovative for HPC users:
  - Tailored for HPC (Systems)
    - Scalable
    - Convenient: Visual, Specific, Recordable, Hardware Aware, Easy to Use
- The New Tools (@github.com/tacc)
  - **core_usage**
  - **show_affinity**
  - **amask**
- Conclusion and future work

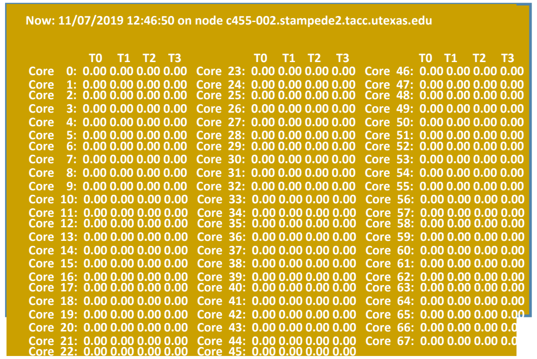
core_usage

- Displays load (%) for all logical processors (“cores”)

- Uses direct counts from /proc/stat (differences)

- GUI or Text-GUI (X11/default or ncurses)





```
Now: 11/07/2019 12:46:50 on node c455-002.stampede2.tacc.utexas.edu

                T0   T1   T2   T3                  T0   T1   T2   T3                  T0   T1   T2   T3
Core  0:  0.00 0.00 0.00 0.00   Core 23:  0.00 0.00 0.00 0.00   Core 46:  0.00 0.00 0.00 0.00
Core  1:  0.00 0.00 0.00 0.00   Core 24:  0.00 0.00 0.00 0.00   Core 47:  0.00 0.00 0.00 0.00
Core  2:  0.00 0.00 0.00 0.00   Core 25:  0.00 0.00 0.00 0.00   Core 48:  0.00 0.00 0.00 0.00
Core  3:  0.00 0.00 0.00 0.00   Core 26:  0.00 0.00 0.00 0.00   Core 49:  0.00 0.00 0.00 0.00
Core  4:  0.00 0.00 0.00 0.00   Core 27:  0.00 0.00 0.00 0.00   Core 50:  0.00 0.00 0.00 0.00
Core  5:  0.00 0.00 0.00 0.00   Core 28:  0.00 0.00 0.00 0.00   Core 51:  0.00 0.00 0.00 0.00
Core  6:  0.00 0.00 0.00 0.00   Core 29:  0.00 0.00 0.00 0.00   Core 52:  0.00 0.00 0.00 0.00
Core  7:  0.00 0.00 0.00 0.00   Core 30:  0.00 0.00 0.00 0.00   Core 53:  0.00 0.00 0.00 0.00
Core  8:  0.00 0.00 0.00 0.00   Core 31:  0.00 0.00 0.00 0.00   Core 54:  0.00 0.00 0.00 0.00
Core  9:  0.00 0.00 0.00 0.00   Core 32:  0.00 0.00 0.00 0.00   Core 55:  0.00 0.00 0.00 0.00
Core 10:  0.00 0.00 0.00 0.00   Core 33:  0.00 0.00 0.00 0.00   Core 56:  0.00 0.00 0.00 0.00
Core 11:  0.00 0.00 0.00 0.00   Core 34:  0.00 0.00 0.00 0.00   Core 57:  0.00 0.00 0.00 0.00
Core 12:  0.00 0.00 0.00 0.00   Core 35:  0.00 0.00 0.00 0.00   Core 58:  0.00 0.00 0.00 0.00
Core 13:  0.00 0.00 0.00 0.00   Core 36:  0.00 0.00 0.00 0.00   Core 59:  0.00 0.00 0.00 0.00
Core 14:  0.00 0.00 0.00 0.00   Core 37:  0.00 0.00 0.00 0.00   Core 60:  0.00 0.00 0.00 0.00
Core 15:  0.00 0.00 0.00 0.00   Core 38:  0.00 0.00 0.00 0.00   Core 61:  0.00 0.00 0.00 0.00
Core 16:  0.00 0.00 0.00 0.00   Core 39:  0.00 0.00 0.00 0.00   Core 62:  0.00 0.00 0.00 0.00
Core 17:  0.00 0.00 0.00 0.00   Core 40:  0.00 0.00 0.00 0.00   Core 63:  0.00 0.00 0.00 0.00
Core 18:  0.00 0.00 0.00 0.00   Core 41:  0.00 0.00 0.00 0.00   Core 64:  0.00 0.00 0.00 0.00
Core 19:  0.00 0.00 0.00 0.00   Core 42:  0.00 0.00 0.00 0.00   Core 65:  0.00 0.00 0.00 0.00
Core 20:  0.00 0.00 0.00 0.00   Core 43:  0.00 0.00 0.00 0.00   Core 66:  0.00 0.00 0.00 0.0
Core 21:  0.00 0.00 0.00 0.00   Core 44:  0.00 0.00 0.00 0.00   Core 67:  0.00 0.00 0.00 0.0
Core 22:  0.00 0.00 0.00 0.00   Core 45:  0.00 0.00 0.00 0.00
```

Syntax

core_usage [sample_period] [txt]

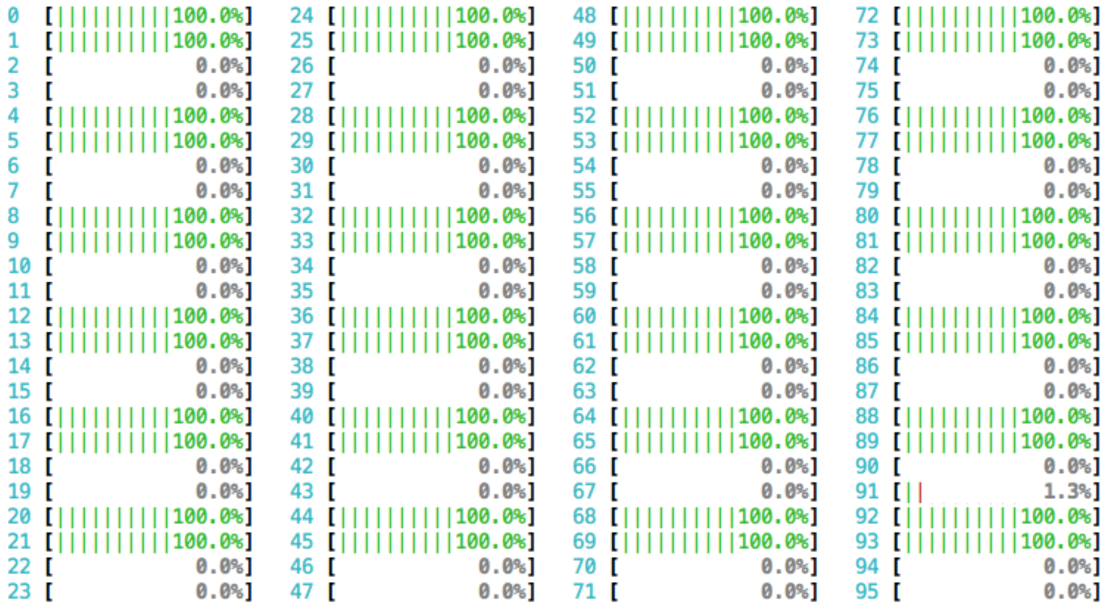
core_usage (GUI)

- Shows Loads
  - so does *htop*, *top*, etc., but
  - Histogram rather than bar graphs

Skylake
HyperThreaded, **48 cores, 96 HWT**

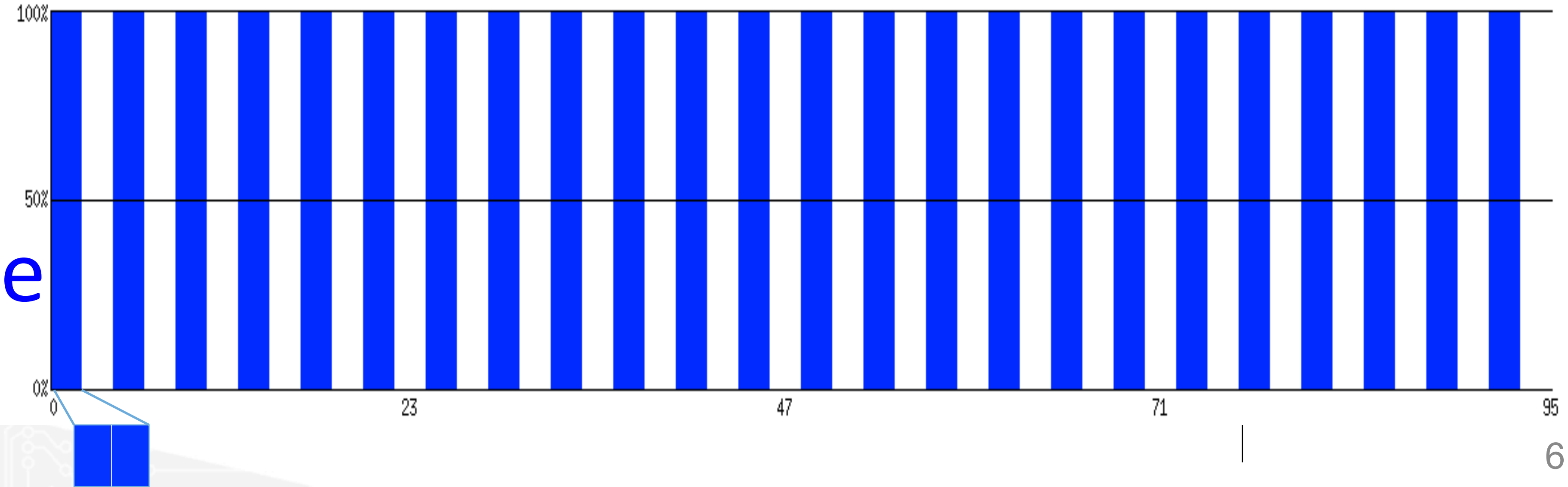
8 MPI tasks X 6 Threads
48 processes & threads
OMP_PLACES=threads
OMP_PROC_BIND=close

htop



```
0  [||||||||||100.0%]   24 [||||||||||100.0%]   48 [||||||||||100.0%]   72 [||||||||||100.0%]
1  [||||||||||100.0%]   25 [||||||||||100.0%]   49 [||||||||||100.0%]   73 [||||||||||100.0%]
2  [            0.0%]   26 [            0.0%]   50 [            0.0%]   74 [            0.0%]
3  [            0.0%]   27 [            0.0%]   51 [            0.0%]   75 [            0.0%]
4  [||||||||||100.0%]   28 [||||||||||100.0%]   52 [||||||||||100.0%]   76 [||||||||||100.0%]
5  [||||||||||100.0%]   29 [||||||||||100.0%]   53 [||||||||||100.0%]   77 [||||||||||100.0%]
6  [            0.0%]   30 [            0.0%]   54 [            0.0%]   78 [            0.0%]
7  [            0.0%]   31 [            0.0%]   55 [            0.0%]   79 [            0.0%]
8  [||||||||||100.0%]   32 [||||||||||100.0%]   56 [||||||||||100.0%]   80 [||||||||||100.0%]
9  [||||||||||100.0%]   33 [||||||||||100.0%]   57 [||||||||||100.0%]   81 [||||||||||100.0%]
10 [            0.0%]   34 [            0.0%]   58 [            0.0%]   82 [            0.0%]
11 [            0.0%]   35 [            0.0%]   59 [            0.0%]   83 [            0.0%]
12 [||||||||||100.0%]   36 [||||||||||100.0%]   60 [||||||||||100.0%]   84 [||||||||||100.0%]
13 [||||||||||100.0%]   37 [||||||||||100.0%]   61 [||||||||||100.0%]   85 [||||||||||100.0%]
14 [            0.0%]   38 [            0.0%]   62 [            0.0%]   86 [            0.0%]
15 [            0.0%]   39 [            0.0%]   63 [            0.0%]   87 [            0.0%]
16 [||||||||||100.0%]   40 [||||||||||100.0%]   64 [||||||||||100.0%]   88 [||||||||||100.0%]
17 [||||||||||100.0%]   41 [||||||||||100.0%]   65 [||||||||||100.0%]   89 [||||||||||100.0%]
18 [            0.0%]   42 [            0.0%]   66 [            0.0%]   90 [            0.0%]
19 [            0.0%]   43 [            0.0%]   67 [            0.0%]   91 [||          1.3%]
20 [||||||||||100.0%]   44 [||||||||||100.0%]   68 [||||||||||100.0%]   92 [||||||||||100.0%]
21 [||||||||||100.0%]   45 [||||||||||100.0%]   69 [||||||||||100.0%]   93 [||||||||||100.0%]
22 [            0.0%]   46 [            0.0%]   70 [            0.0%]   94 [            0.0%]
23 [            0.0%]   47 [            0.0%]   71 [            0.0%]   95 [            0.0%]
```

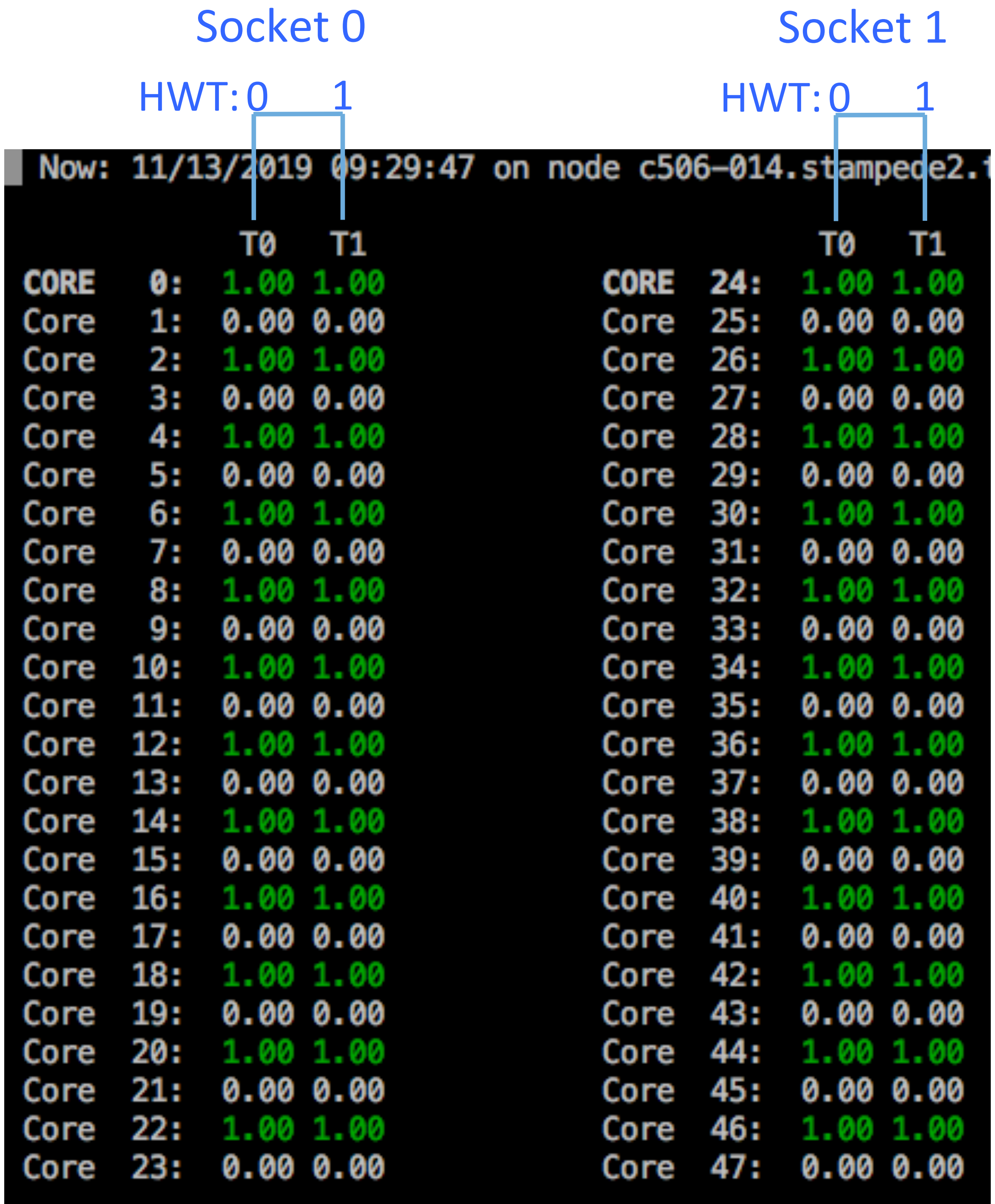
core_usage



core_usage (txt)

- Shows Loads
  - so does *htop*, *top*, etc., but
  - Hardware Aware!

8 MPI tasks X 6 Threads
4 domains
OMP_PLACES=threads
OMP_PROC_BIND=close



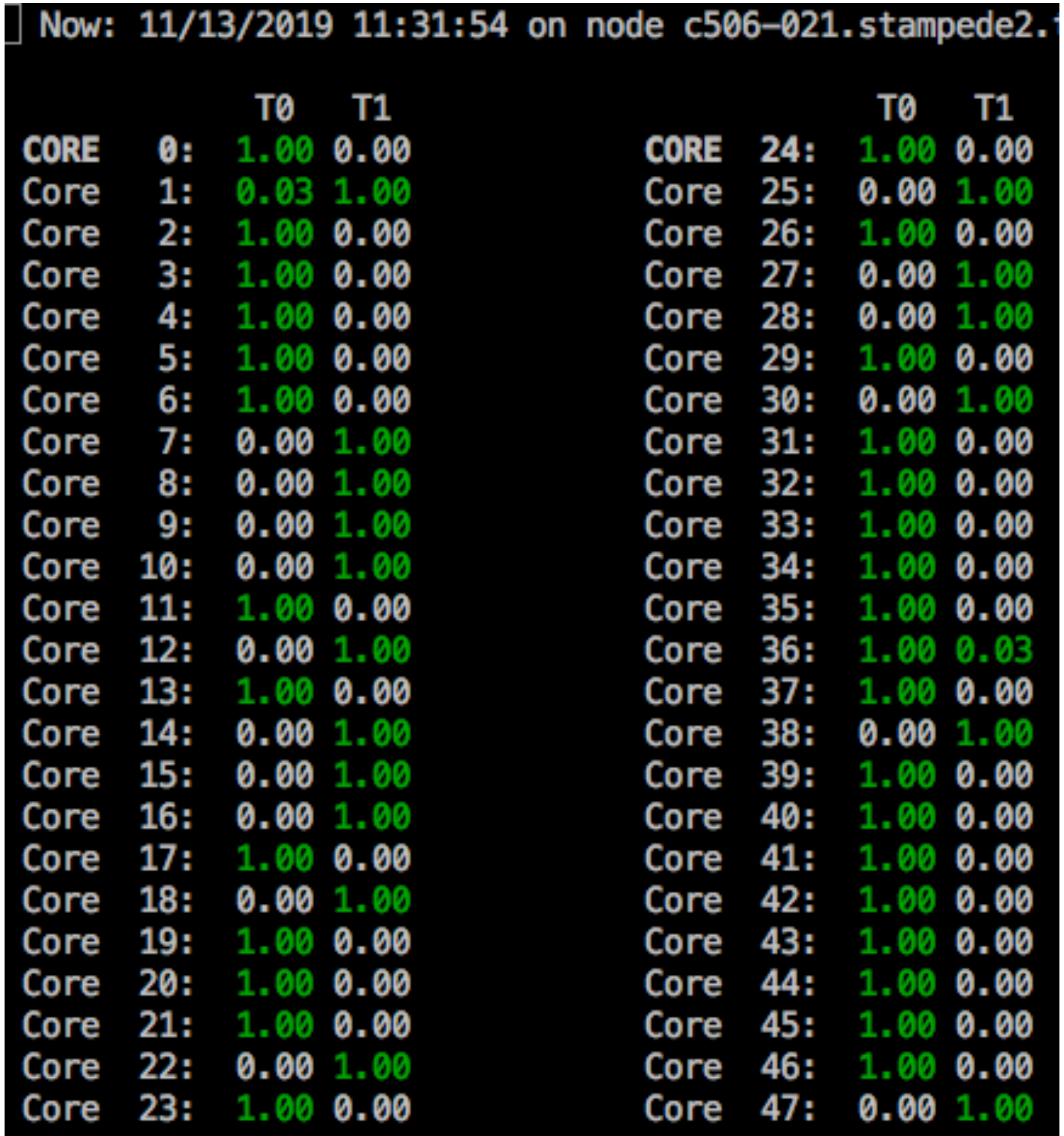
```
Now: 11/13/2019 09:29:47 on node c506-014.stampede2.t

              T0   T1                        T0   T1
CORE    0:  1.00 1.00         CORE   24:  1.00 1.00
Core    1:  0.00 0.00         Core   25:  0.00 0.00
Core    2:  1.00 1.00         Core   26:  1.00 1.00
Core    3:  0.00 0.00         Core   27:  0.00 0.00
Core    4:  1.00 1.00         Core   28:  1.00 1.00
Core    5:  0.00 0.00         Core   29:  0.00 0.00
Core    6:  1.00 1.00         Core   30:  1.00 1.00
Core    7:  0.00 0.00         Core   31:  0.00 0.00
Core    8:  1.00 1.00         Core   32:  1.00 1.00
Core    9:  0.00 0.00         Core   33:  0.00 0.00
Core   10:  1.00 1.00         Core   34:  1.00 1.00
Core   11:  0.00 0.00         Core   35:  0.00 0.00
Core   12:  1.00 1.00         Core   36:  1.00 1.00
Core   13:  0.00 0.00         Core   37:  0.00 0.00
Core   14:  1.00 1.00         Core   38:  1.00 1.00
Core   15:  0.00 0.00         Core   39:  0.00 0.00
Core   16:  1.00 1.00         Core   40:  1.00 1.00
Core   17:  0.00 0.00         Core   41:  0.00 0.00
Core   18:  1.00 1.00         Core   42:  1.00 1.00
Core   19:  0.00 0.00         Core   43:  0.00 0.00
Core   20:  1.00 1.00         Core   44:  1.00 1.00
Core   21:  0.00 0.00         Core   45:  0.00 0.00
Core   22:  1.00 1.00         Core   46:  1.00 1.00
Core   23:  0.00 0.00         Core   47:  0.00 0.00
```

core_usage (txt)

- Shows Loads
  - so does *htop*, *top*, etc., but
  - Hardware Aware!

8 MPI tasks X 6 Threads
4 domains
OMP_PLACES=**cores**
OMP_PROC_BIND=close



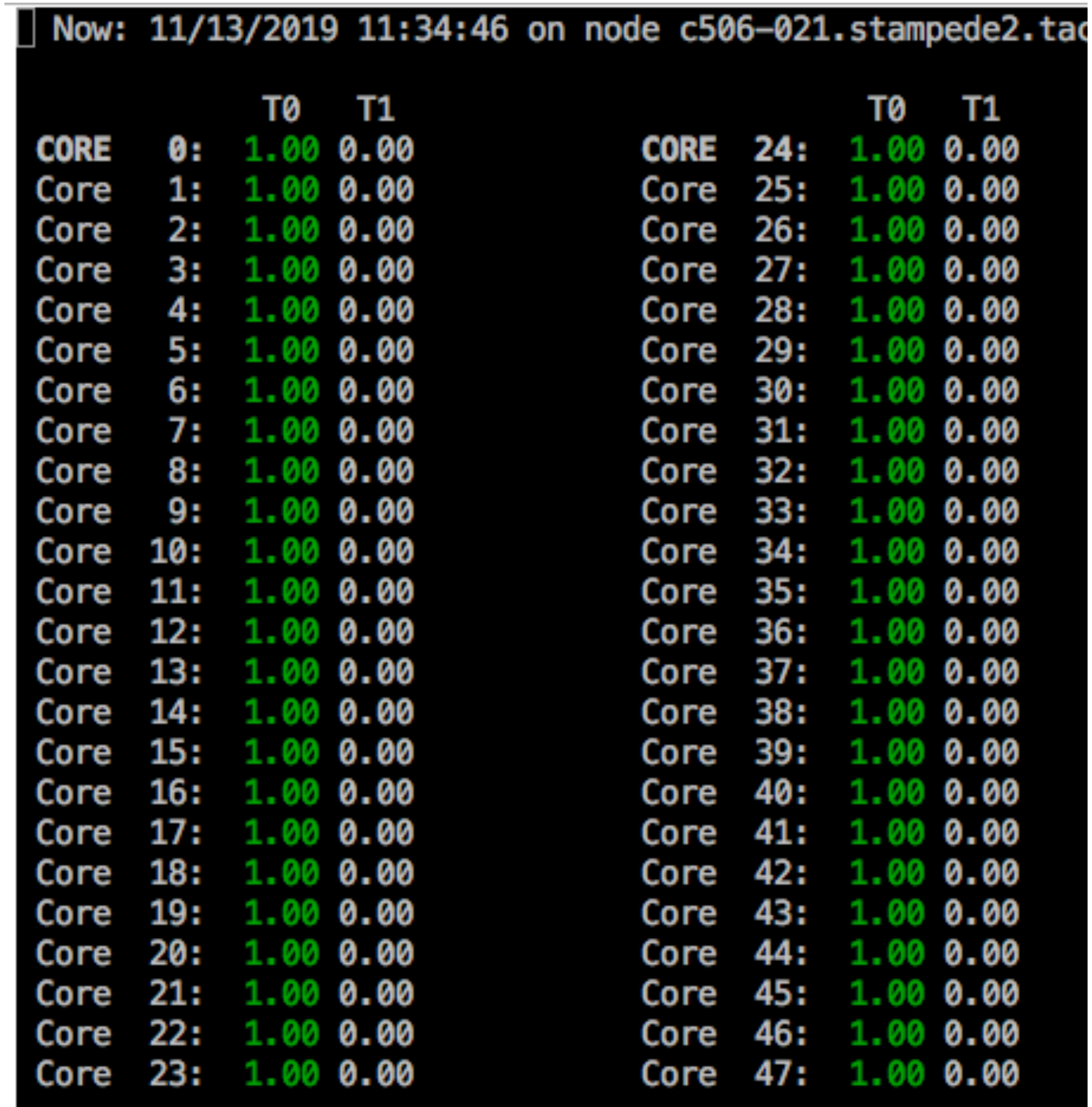
```
Now: 11/13/2019 11:31:54 on node c506-021.stampede2.

               T0   T1                        T0   T1
CORE     0:  1.00 0.00           CORE  24:  1.00 0.00
Core     1:  0.03 1.00           Core  25:  0.00 1.00
Core     2:  1.00 0.00           Core  26:  1.00 0.00
Core     3:  1.00 0.00           Core  27:  0.00 1.00
Core     4:  1.00 0.00           Core  28:  0.00 1.00
Core     5:  1.00 0.00           Core  29:  1.00 0.00
Core     6:  1.00 0.00           Core  30:  0.00 1.00
Core     7:  0.00 1.00           Core  31:  1.00 0.00
Core     8:  0.00 1.00           Core  32:  1.00 0.00
Core     9:  0.00 1.00           Core  33:  1.00 0.00
Core    10:  0.00 1.00           Core  34:  1.00 0.00
Core    11:  1.00 0.00           Core  35:  1.00 0.00
Core    12:  0.00 1.00           Core  36:  1.00 0.03
Core    13:  1.00 0.00           Core  37:  1.00 0.00
Core    14:  0.00 1.00           Core  38:  0.00 1.00
Core    15:  0.00 1.00           Core  39:  1.00 0.00
Core    16:  0.00 1.00           Core  40:  1.00 0.00
Core    17:  1.00 0.00           Core  41:  1.00 0.00
Core    18:  0.00 1.00           Core  42:  1.00 0.00
Core    19:  1.00 0.00           Core  43:  1.00 0.00
Core    20:  1.00 0.00           Core  44:  1.00 0.00
Core    21:  1.00 0.00           Core  45:  1.00 0.00
Core    22:  0.00 1.00           Core  46:  1.00 0.00
Core    23:  1.00 0.00           Core  47:  0.00 1.00
```

core_usage (txt)

- Shows Loads
  - so does *htop*, *top*, etc., but
  - Hardware Aware!

8 MPI tasks X 6 Threads
4 domains
OMP_PLACES=**threads**
OMP_PROC_BIND=spread



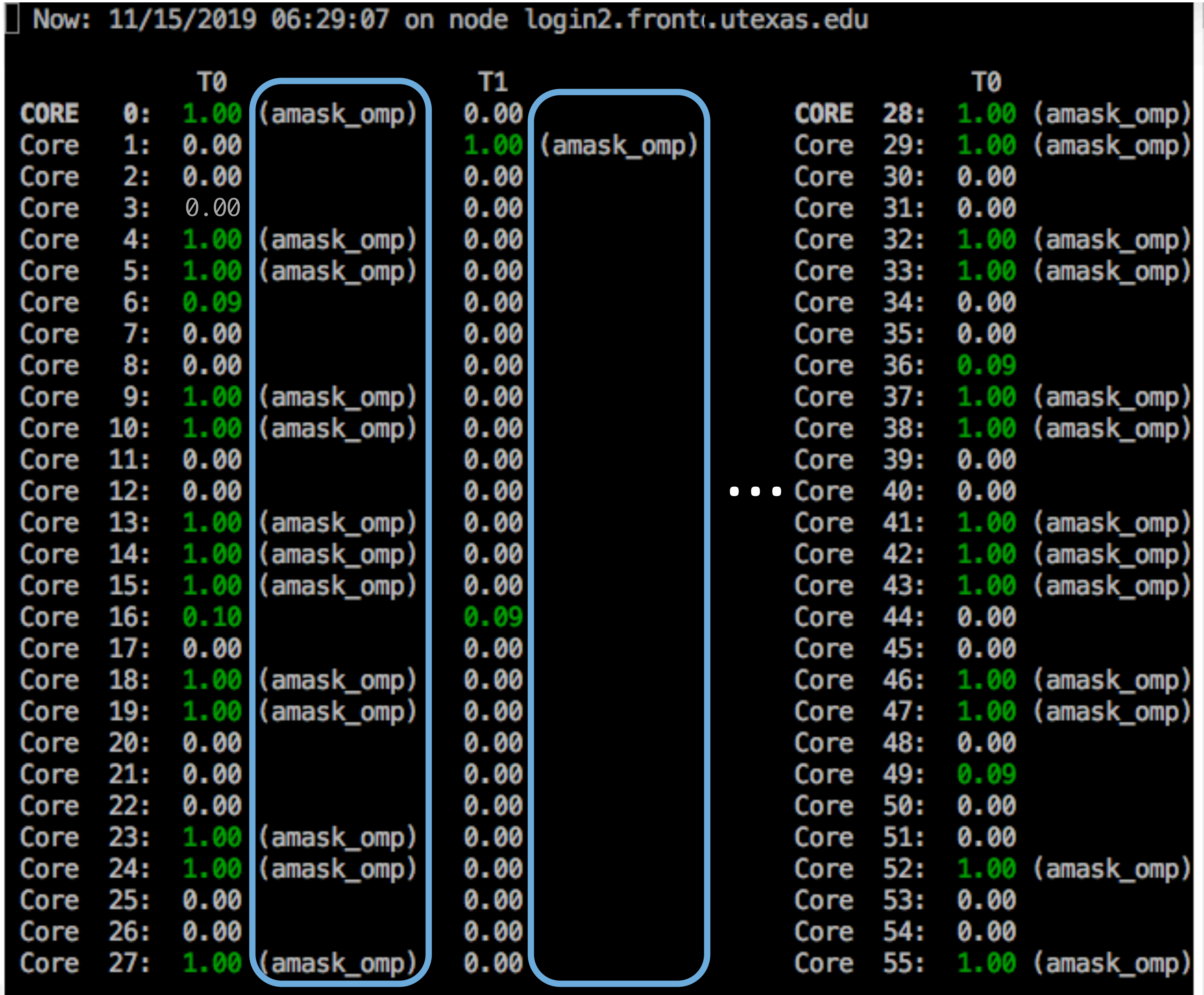
```
Now: 11/13/2019 11:34:46 on node c506-021.stampede2.tac

               T0   T1                            T0   T1
CORE     0:  1.00 0.00                CORE   24:  1.00 0.00
Core     1:  1.00 0.00                Core   25:  1.00 0.00
Core     2:  1.00 0.00                Core   26:  1.00 0.00
Core     3:  1.00 0.00                Core   27:  1.00 0.00
Core     4:  1.00 0.00                Core   28:  1.00 0.00
Core     5:  1.00 0.00                Core   29:  1.00 0.00
Core     6:  1.00 0.00                Core   30:  1.00 0.00
Core     7:  1.00 0.00                Core   31:  1.00 0.00
Core     8:  1.00 0.00                Core   32:  1.00 0.00
Core     9:  1.00 0.00                Core   33:  1.00 0.00
Core    10:  1.00 0.00                Core   34:  1.00 0.00
Core    11:  1.00 0.00                Core   35:  1.00 0.00
Core    12:  1.00 0.00                Core   36:  1.00 0.00
Core    13:  1.00 0.00                Core   37:  1.00 0.00
Core    14:  1.00 0.00                Core   38:  1.00 0.00
Core    15:  1.00 0.00                Core   39:  1.00 0.00
Core    16:  1.00 0.00                Core   40:  1.00 0.00
Core    17:  1.00 0.00                Core   41:  1.00 0.00
Core    18:  1.00 0.00                Core   42:  1.00 0.00
Core    19:  1.00 0.00                Core   43:  1.00 0.00
Core    20:  1.00 0.00                Core   44:  1.00 0.00
Core    21:  1.00 0.00                Core   45:  1.00 0.00
Core    22:  1.00 0.00                Core   46:  1.00 0.00
Core    23:  1.00 0.00                Core   47:  1.00 0.00
```

core_usage (txt)

New Version Shows

Application Name

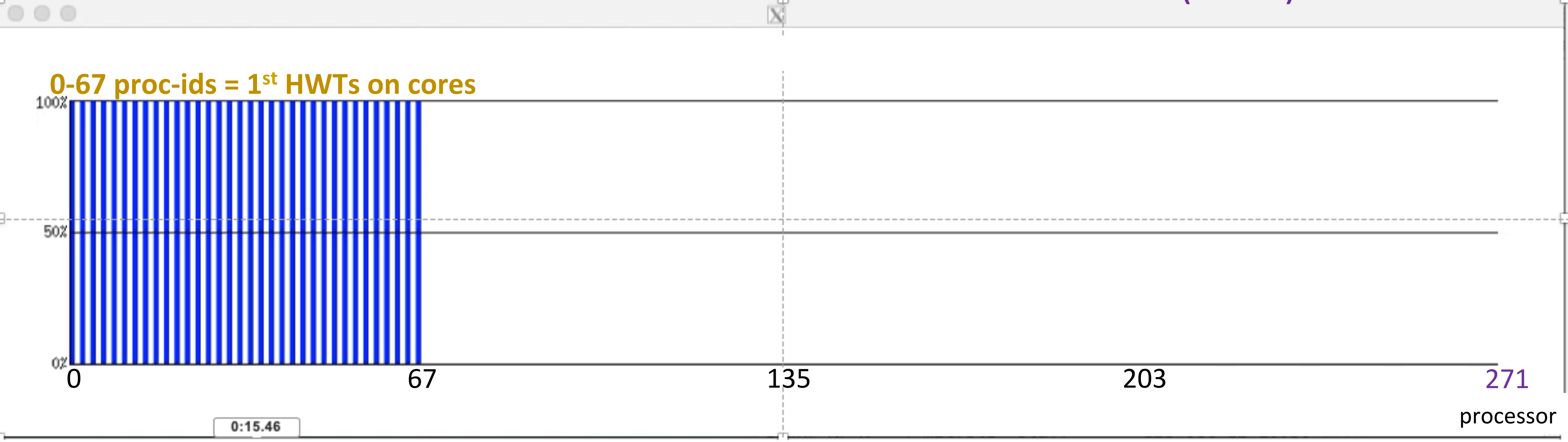


```
Now: 11/15/2019 06:29:07 on node login2.front.utexas.edu

          T0                 T1                              T0
CORE  0: 1.00 (amask_omp)   0.00                   CORE 28: 1.00 (amask_omp)
Core  1: 0.00               1.00 (amask_omp)       Core 29: 1.00 (amask_omp)
Core  2: 0.00               0.00                   Core 30: 0.00
Core  3: 0.00               0.00                   Core 31: 0.00
Core  4: 1.00 (amask_omp)   0.00                   Core 32: 1.00 (amask_omp)
Core  5: 1.00 (amask_omp)   0.00                   Core 33: 1.00 (amask_omp)
Core  6: 0.09               0.00                   Core 34: 0.00
Core  7: 0.00               0.00                   Core 35: 0.00
Core  8: 0.00               0.00                   Core 36: 0.09
Core  9: 1.00 (amask_omp)   0.00                   Core 37: 1.00 (amask_omp)
Core 10: 1.00 (amask_omp)   0.00                   Core 38: 1.00 (amask_omp)
Core 11: 0.00               0.00                   Core 39: 0.00
Core 12: 0.00               0.00               ... Core 40: 0.00
Core 13: 1.00 (amask_omp)   0.00                   Core 41: 1.00 (amask_omp)
Core 14: 1.00 (amask_omp)   0.00                   Core 42: 1.00 (amask_omp)
Core 15: 1.00 (amask_omp)   0.00                   Core 43: 1.00 (amask_omp)
Core 16: 0.10               0.09                   Core 44: 0.00
Core 17: 0.00               0.00                   Core 45: 0.00
Core 18: 1.00 (amask_omp)   0.00                   Core 46: 1.00 (amask_omp)
Core 19: 1.00 (amask_omp)   0.00                   Core 47: 1.00 (amask_omp)
Core 20: 0.00               0.00                   Core 48: 0.00
Core 21: 0.00               0.00                   Core 49: 0.09
Core 22: 0.00               0.00                   Core 50: 0.00
Core 23: 1.00 (amask_omp)   0.00                   Core 51: 0.00
Core 24: 1.00 (amask_omp)   0.00                   Core 52: 1.00 (amask_omp)
Core 25: 0.00               0.00                   Core 53: 0.00
Core 26: 0.00               0.00                   Core 54: 0.00
Core 27: 1.00 (amask_omp)   0.00                   Core 55: 1.00 (amask_omp)
```

core_usage observation

KNL 4thrds X 68 Cores : 272 “logical” processors
272 “Hardware Threads (HWT)”



0-67 proc-ids = 1st HWTs on cores

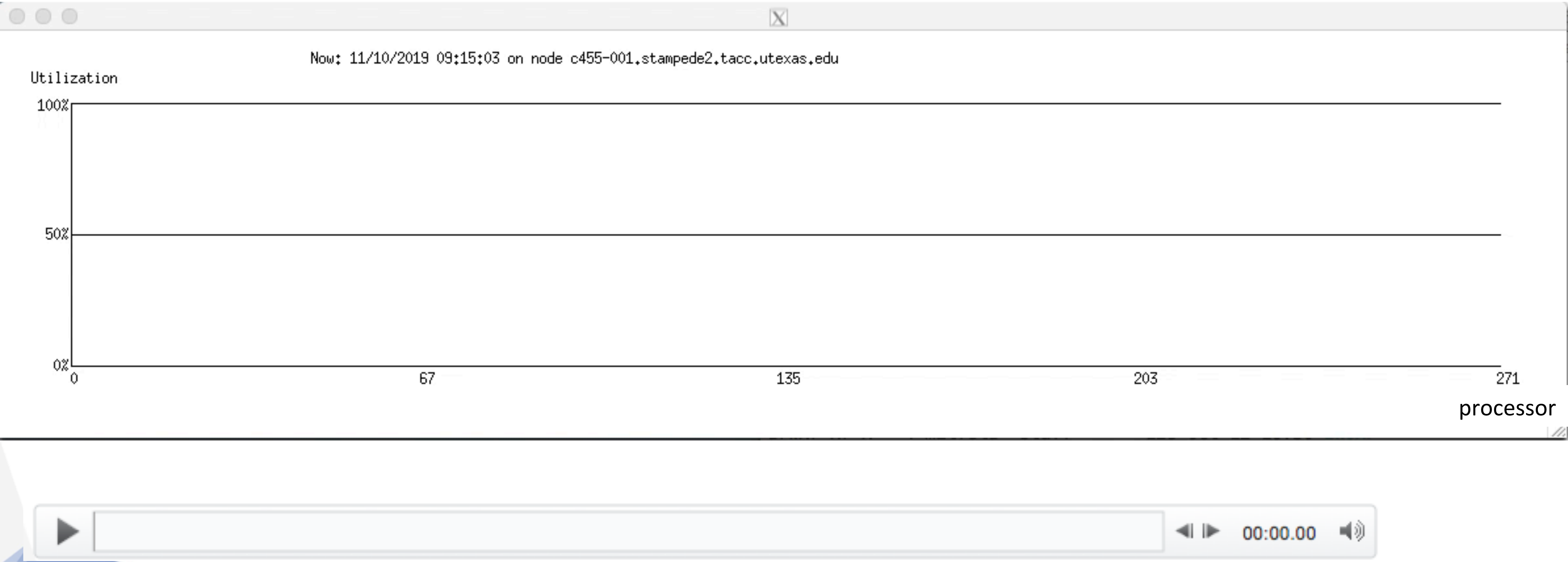
34 OpenMP tasks

Affinity: OMP_PLACES=cores

OMP_PROC_BIND=spread

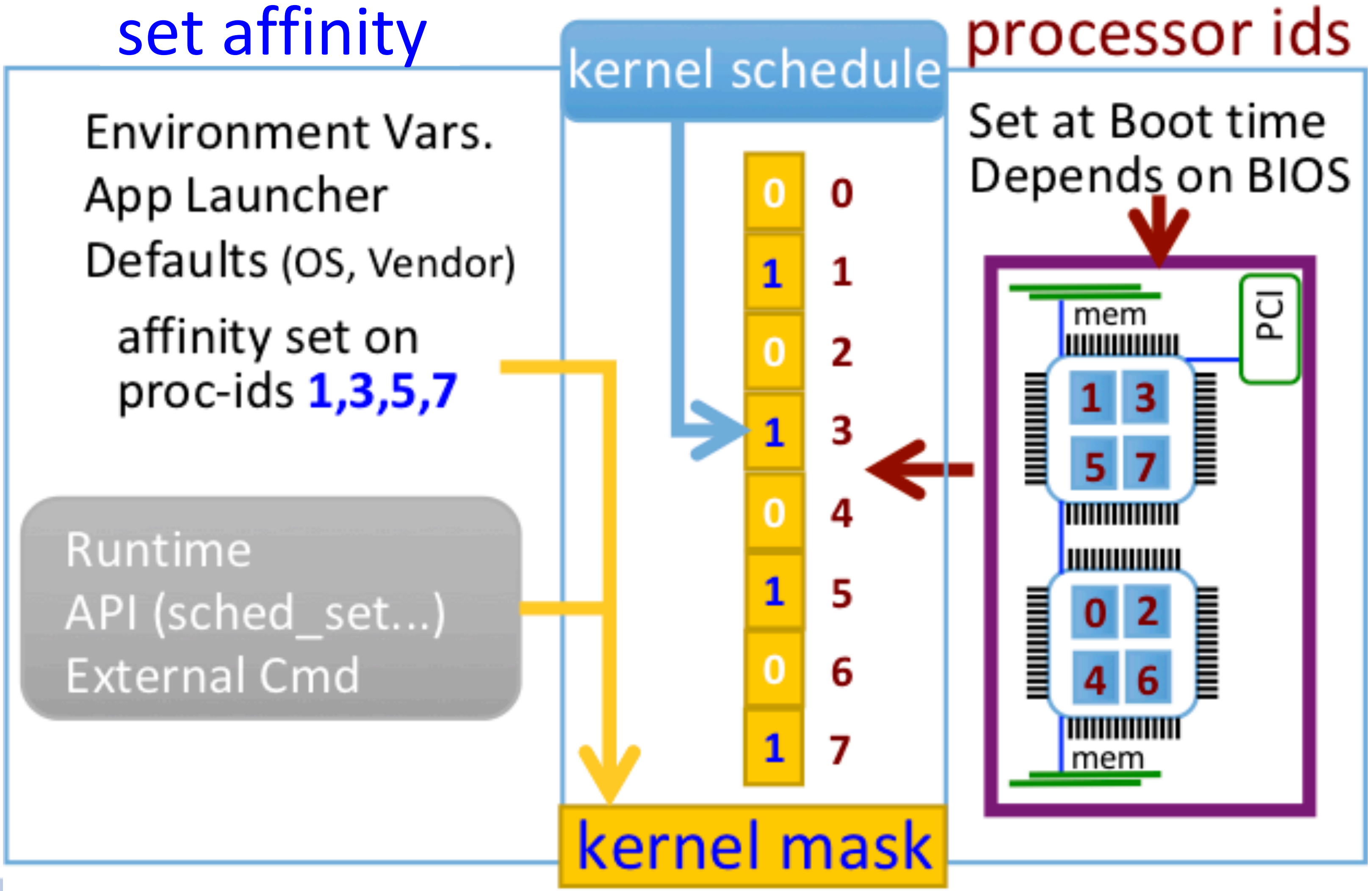
core_usage movie

KNL: 34 OpenMP tasks
Affinity: OMP_PLACES=cores
OMP_PROC_BIND=spread



Now: 11/10/2019 09:15:03 on node c455-001.stampede2.tacc.utexas.edu

Process affinity



show_affinity

- Displays affinity information→ pid, cmd, tid, affinity(proc-ids)

- Inspects only user's current (time-consuming) processes in /proc
- Display "core" affinity of each individual process/thread of app

- Enumerate All User's running processes/threads

Syntax

```
show_affinity [all]
```

Retrieving Process Affinity

- Taskset: ***taskset -a -p*** <pid>
  - use *ps -lfu $USER* to get pids
  - MPI -- Loop over pids
  - OpenMP– all threads: use -a (get LWP + Hex map)
- Vendor/Implementation:
  - IMPI: include I_MPI_DEBUG=4
  - OpenMP: export OMP_DISPLAY_AFFINITY=TRUE
- Utilities:
  - htop, etc.
  - show_affinity

taskset

For those who live and breathe hex!

```
$ ps -lfu $USER | grep amask_omp
F S UID        PID      PPID    PRI ...   TIME        CMD         ←added for clarity
0 R milfeld   81634  80527 99  ...    00:01:35 amask_omp -w 120

$ taskset -a -p 81634
pid 81634's current affinity mask: 5
pid 81635's current affinity mask: 14
pid 81636's current affinity mask: 50
pid 81637's current affinity mask: 140
pid 81638's current affinity mask: 500
pid 81639's current affinity mask: 1400
pid 81640's current affinity mask: 5000
pid 81641's current affinity mask: 14000
```

show_affinity

KNL 68 cores, 272 HWT

export I_MPI_DEBUG=4
mpirun –np 8 app

```
[0] MPI startup(): 0       161195   c455-011.stampede2.tacc.utexas.edu
{0,1,2,3,4,5,6,7,8,68,69,70,71,72,73,74,75,76,136,137,138,139,140,141,142,143,204
                                                          ,205,206,207,208,209,210,211}
[0] MPI startup(): 1       161196   c455-011.stampede2.tacc.utexas.edu
{9,10,11,12,13,14,15,16,77,78,79,80,81,82,83,84,144,145,146,147,148,149,150,151,1
                                                          52,212,213,214,215,216,217,218,219,220}
[0] MPI startup(): 2       161197   c455-011.stampede2.tacc.utexas.edu
{17,18,19,20,21,22,23,24,25,85,86,87,88,89,90,91,92,93,153,154,155,156,157,158,15
                                                          9,160,221,222,223,224,225,226,227,228}
[0] MPI startup(): 3       161198   c455-011.stampede2.tacc.utexas.edu
{26,27,28,29,30,31,32,33,94,95,96,97,98,99,100,101,161,162,163,164,165,166,167,16
                                                          8,169,229,230,231,232,233,234,235,236,237}
[0] MPI startup(): 4       161199   c455-011.stampede2.tacc.utexas.edu
{34,35,36,37,38,39,40,41,42,102,103,104,105,106,107,108,109,110,170,171,172,173,1
                                                          74,175,176,177,238,239,240,241,242,243,244,245}
[0] MPI startup(): 5       161200   c455-011.stampede2.tacc.utexas.edu
{43,44,45,46,47,48,49,50,111,112,113,114,115,116,117,118,178,179,180,181,182,183,
                                                          184,185,186,246,247,248,249,250,251,252,253,254}
[0] MPI startup(): 6       161201   c455-011.stampede2.tacc.utexas.edu
{51,52,53,54,55,56,57,58,59,119,120,121,122,123,124,125,126,127,187,188,189,190,1
                                                          91,192,193,194,255,256,257,258,259,260,261,262}
[0] MPI startup(): 7       161203   c455-011.stampede2.tacc.utexas.edu
{60,61,62,63,64,65,66,67,128,129,130,131,132,133,134,135,195,196,197,198,199,200,
                                                          201,202,203,263,264,265,266,267,268,269,270,271}
```

Displayed by runtime.
`I_MPI_DEBUG` must be set
prior to execution. [Not portable.]

$ mpirun -np 8 app   ***$ show_affinity***

```
 pid      Exe_Name        tid      Affinity
161195  amask_mpi        161195   0-8,68-76,136-143,204-211
161196  amask_mpi        161196   9-16,77-84,144-152,212-220
161197  amask_mpi        161197   17-25,85-93,153-160,221-228
161198  amask_mpi        161198   26-33,94-101,161-169,229-237
161199  amask_mpi        161199   34-42,102-110,170-177,238-245
161200  amask_mpi        161200   43-50,111-118,178-186,246-254
161201  amask_mpi        161201   51-59,119-127,187-194,255-262
161202  amask_mpi        161202   60-67,128-135,195-203,263-271
```

show_affinity is a detached process.
Jump on node and run anytime!

show_affinity

Cascade Lake, 56 cores, HWT=112

export OMP_NUM_THREADS=8 OMP_PROC_BIND=TRUE
export OMP_DISPLAY_AFFINITY=TRUE

```
$ export OMP_DISPLAY_AFFINITY=TRUE
$ app
```

```
OMP: pid 219800 tid 219800 thread 0 bound to OS proc set {0,56}
OMP: pid 219800 tid 219807 thread 7 bound to OS proc set {45,101}
OMP: pid 219800 tid 219804 thread 4 bound to OS proc set {1,57}
OMP: pid 219800 tid 219803 thread 3 bound to OS proc set {44,100}
OMP: pid 219800 tid 219801 thread 1 bound to OS proc set {16,72}
OMP: pid 219800 tid 219805 thread 5 bound to OS proc set {17,73}
OMP: pid 219800 tid 219802 thread 2 bound to OS proc set {28,84}
OMP: pid 219800 tid 219806 thread 6 bound to OS proc set {29,85}
```

Displayed by runtime at parallel region. `OMP_DISPLAY_AFFINITY` must be set prior to execution.

```
$ app
```

```
$ show_affinity
```

show_affinity is a detached process.
Jump on node and run anytime!
watch -n 1 show_affinity

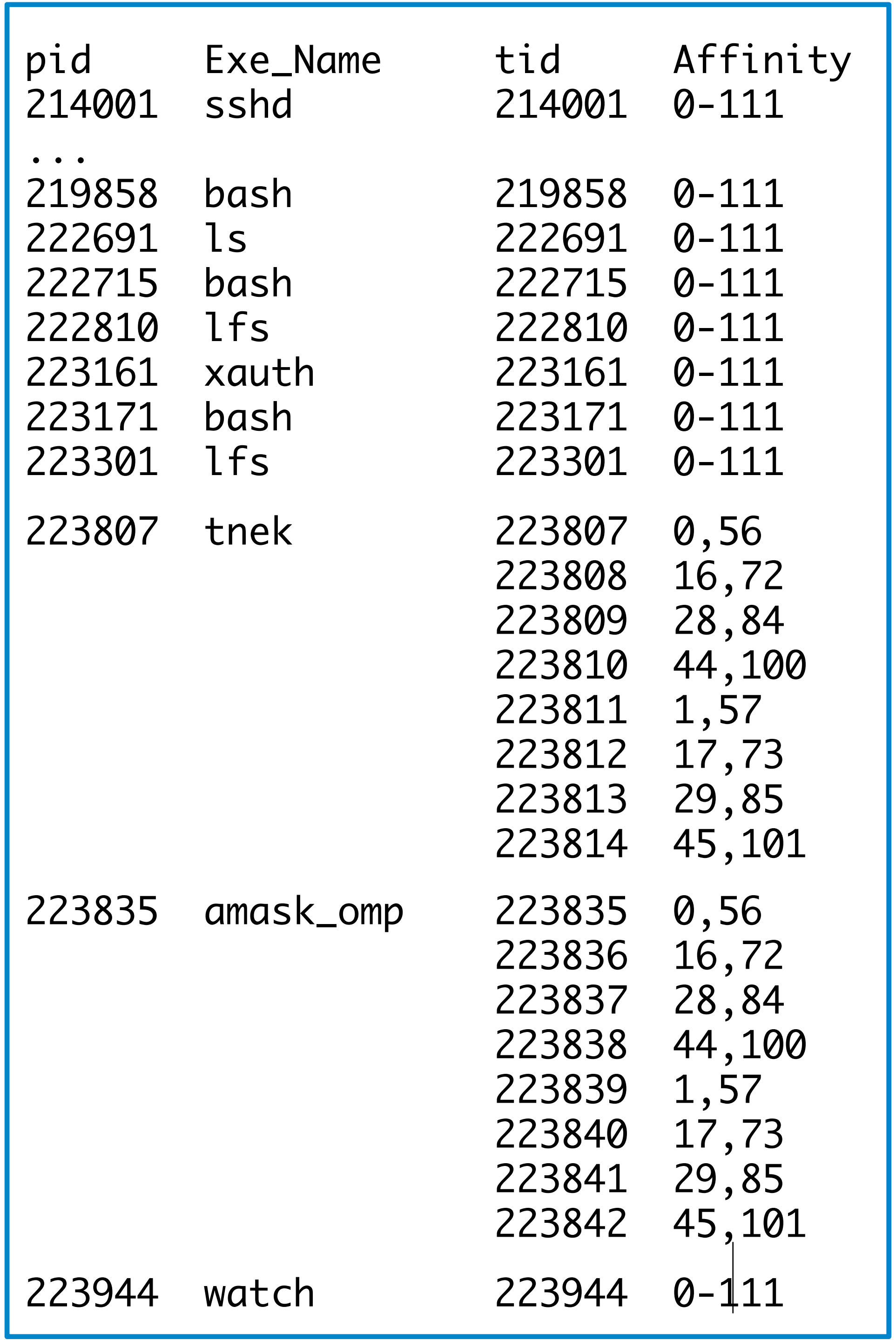
```
pid     Exe_Name        tid     Affinity
220034  amask_omp       220034  0,56
                        220035  16,72
                        220036  28,84
                        220037  44,100
                        220038  1,57
                        220039  17,73
                        220040  29,85
                        220041  45,101
```

show_affinity

- Watch all user-owned processes
    - Thread ids for all user processes
    - Ordered, Easy to determine new processes.

1.) Thread ids for all user processes
2.) Ordered, Easy to determine new processes

```
$ ssh <a_compute_node>
$ watch -n 5 show_affinity all
```



```
pid     Exe_Name      tid     Affinity
214001  sshd          214001  0-111
...
219858  bash          219858  0-111
222691  ls            222691  0-111
222715  bash          222715  0-111
222810  lfs           222810  0-111
223161  xauth         223161  0-111
223171  bash          223171  0-111
223301  lfs           223301  0-111

223807  tnek          223807  0,56
                      223808  16,72
                      223809  28,84
                      223810  44,100
                      223811  1,57
                      223812  17,73
                      223813  29,85
                      223814  45,101

223835  amask_omp     223835  0,56
                      223836  16,72
                      223837  28,84
                      223838  44,100
                      223839  1,57
                      223840  17,73
                      223841  29,85
                      223842  45,101

223944  watch         223944  0-111
```

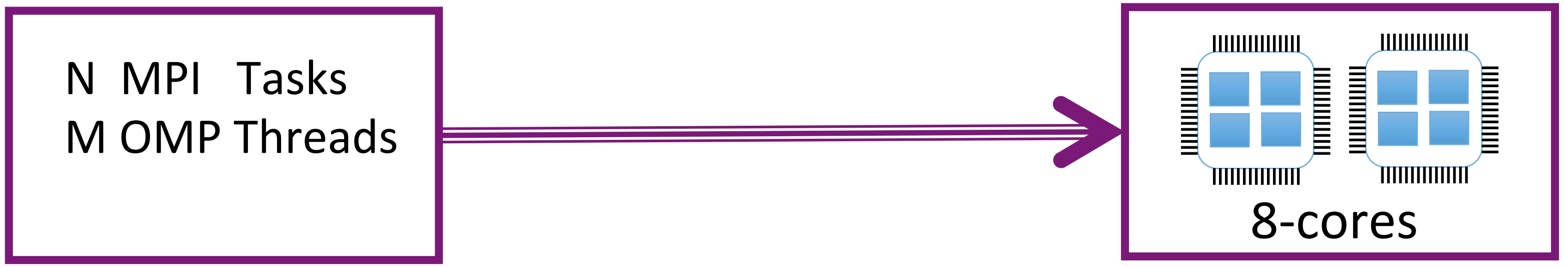
show_affinity

- Summary
  - Detached utility
  - Compact Numbering
  - Automaticly detects loaded *PID* (and *TID*) of User's App.
  - Detects *all* (extraneous) processes

```
pid     Exe_Name      tid     Affinity
214001  sshd          214001  0-111
...
219858  bash          219858  0-111
222691  ls            222691  0-111
222715  bash          222715  0-111
222810  lfs           222810  0-111
223161  xauth         223161  0-111
223171  bash          223171  0-111
223301  lfs           223301  0-111
223807  tnek          223807  0,56
                      223808  16,72
                      223809  28,84
                      223810  44,100
                      223811  1,57
                      223812  17,73
                      223813  29,85
                      223814  45,101
223835  amask_omp     223835  0,56
                      223836  16,72
                      223837  28,84
                      223838  44,100
                      223839  1,57
                      223840  17,73
                      223841  29,85
                      223842  45,101
223944  watch         223944  0-111
```

CPU Affinity -- the mask

processes map onto processors



N MPI Tasks
M OMP Threads

8-cores

MPI rank-num or thread-num

proc-id (core-id)

A kernel bit mask (array of bits) exists for each process

# of bits = # of processors (proc-ids) set bit→ process can run on proc-id (core-id).

	thrd	0	1	2	3	4	5	6	7	proc-id
allow rank/thrd-num 0 to run on cores 0-3 →	0	1	1	1	1					
allow rank/thrd-num 1 to run on cores 4-7 →	1					1	1	1	1	

amask

- Reports comprehensible affinity info for an affinity env.
  - Set affinity, then execute amask_<*type*>   types:{omp, mpi, hybrid}.
  - Can instrument code with amask_<*type*>() argumentless function calls.

Syntax:

amask_<*type*>   [-h] [-w#] [-vk]

help
wait #sec with load
view kernel mask

Usage: module load amask

```
export OMP_<affinity>= ...              →                amask_omp        # for pure OpenMP
export I_MPI_<affinity>= ...            →         mpirun amask_mpi        # for pure MPI
export OMP_<affinity>= ... \            →         mpirun amask_hybrid     # MPI + OpenMP
           I_MPI_<affinity>= ...
```

Viewing Affinity mask with amask

Old Stampede: 16-core

```
export OMP_NUM_THREADS=8  OMP_PROC_BIND=spread
amask_omp
```

```
         0    proc-id     15
thrd     |                 |
   0     1000000000000000
   1     0010000000000000
   2     0000100000000000
   3     0000001000000000
   4     0000000010000000
   5     0000000000100000
   6     0000000000001000
   7     0000000000000010
```

Bit Mask

```
                  proc-id
thrd    |  0       |       10     |
   0    1---------------
   1    --1-------------
   2    ----1-----------
   3    ------1---------
   4    -------1--------
   5    ---------1------
   6    ------------1---
   7    --------------1-
rank    |  0       |       10     |
```

More Readable

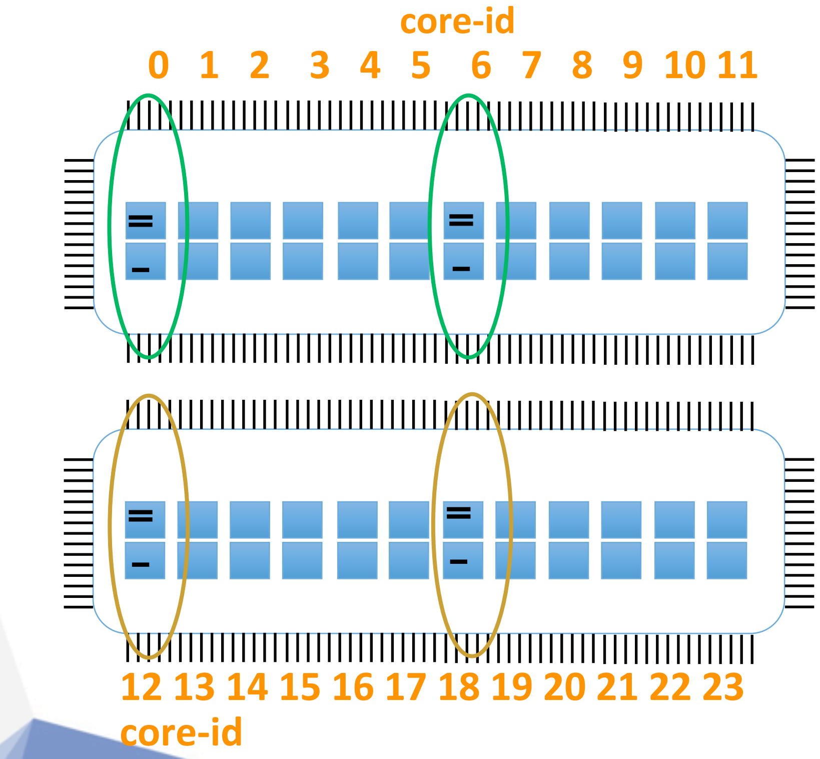
```
                  proc-id
thrd    |  0       |       10     |
   0    0---------------
   1    --2-------------
   2    ----4-----------
   3    ------6---------
   4    --------8-------
   5    ----------0-----
   6    ------------2---
   7    --------------4-
rank    |  0       |       10     |
```

Even More Readable

What about hyper-threading…

Hyper-Threaded systems   2 x 12 cores → 48 hardware threads

```
$ export OMP_NUM_THREADS=4 OMP_PLACES=cores
$ amask_omp
```



view core mask (core-ids)

```
thrd |    0    |   10    |   20   ← core-id
0000 0=======================     HW-thread 0
     0-----------------------     HW-thread 1
0001 ======6=================
     ------6-----------------
0002 ============2===========
     ------------2-----------
0003 ==================8=====
     ------------------8-----
```

What about SMT… on KNL

KNL:
68 cores
4 HWT

spread (default)
Implies
distribution
with a stride
of 4.

```
$ export OMP_NUM_THREADS=17 OMP_PLACES=cores OMP_PROC_BIND=spread
$ amask_omp
```

```
thrd |    0    |   10    |   20    |   30    |   40    |   50    |   60    |
0000 0===================================================================
     0-------------------------------------------------------------------
     0-------------------------------------------------------------------
     0-------------------------------------------------------------------
0001 ====4===============================================================
     ----4---------------------------------------------------------------
     ----4---------------------------------------------------------------
     ----4---------------------------------------------------------------
0002 ========8===========================================================
     --------8-----------------------------------------------------------
     --------8-----------------------------------------------------------
     --------8-----------------------------------------------------------
                 ...
0016 ================================================================4===
     ----------------------------------------------------------------4---
     ----------------------------------------------------------------4---
     ----------------------------------------------------------------4---
```

What about SMT… on KNL

Pure MPI:
mpirun -np 8 amask_mpi

Default Affinity:
Note: Cores 8, 25
42, and 59 share
cores!

```
rank |    0    |   10    |   20    |   30    |   40    |   50    |   60    |
0000 012345678===========================================================
     012345678-----------------------------------------------------------
     01234567------------------------------------------------------------
     01234567------------------------------------------------------------
0001 =========90123456===================================================
     ---------90123456---------------------------------------------------
     --------890123456---------------------------------------------------
     --------890123456---------------------------------------------------
0002 =================789012345==========================================
     -----------------789012345------------------------------------------
     -----------------78901234-------------------------------------------
     -----------------78901234-------------------------------------------
0003 ==========================67890123==================================
     --------------------------67890123----------------------------------
     -------------------------567890123----------------------------------
     -------------------------567890123----------------------------------
                                                ...

0006 ===================================================123456789========
     ---------------------------------------------------123456789--------
     ---------------------------------------------------12345678---------
     ---------------------------------------------------12345678---------
0007 ============================================================01234567
     ------------------------------------------------------------01234567
     -----------------------------------------------------------901234567
     -----------------------------------------------------------901234567
```

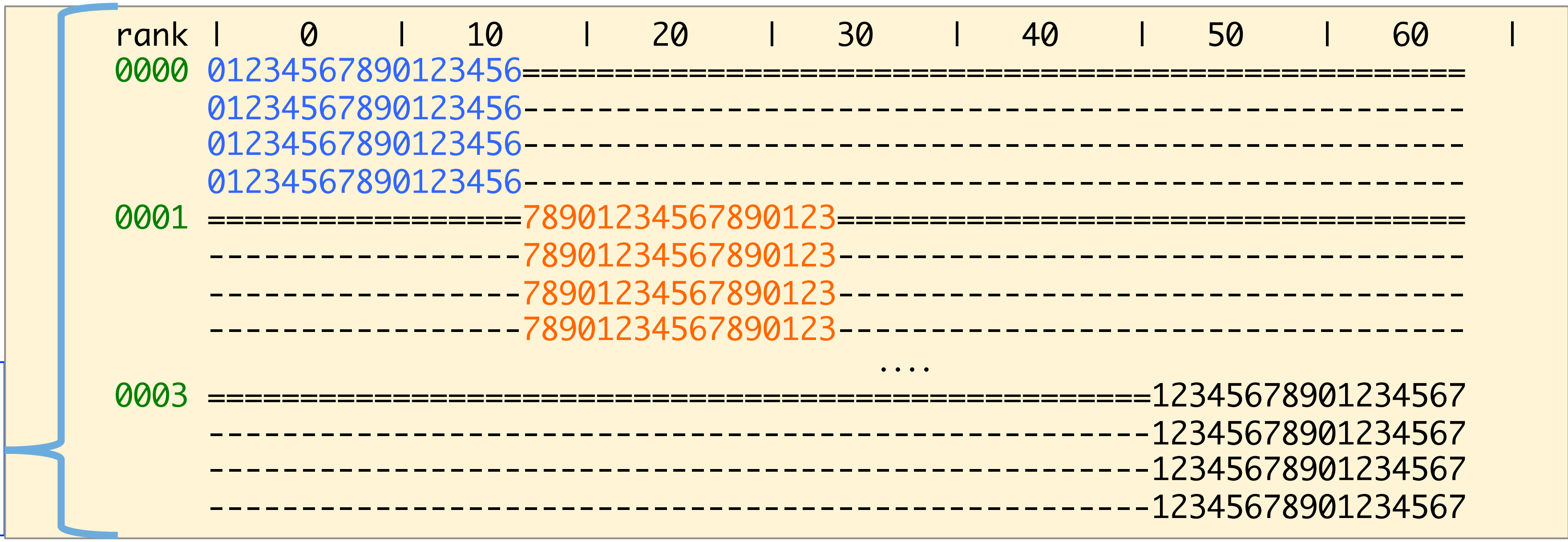
KNL -hybrid

Hybrid:

4 MPI tasks x 17 threads

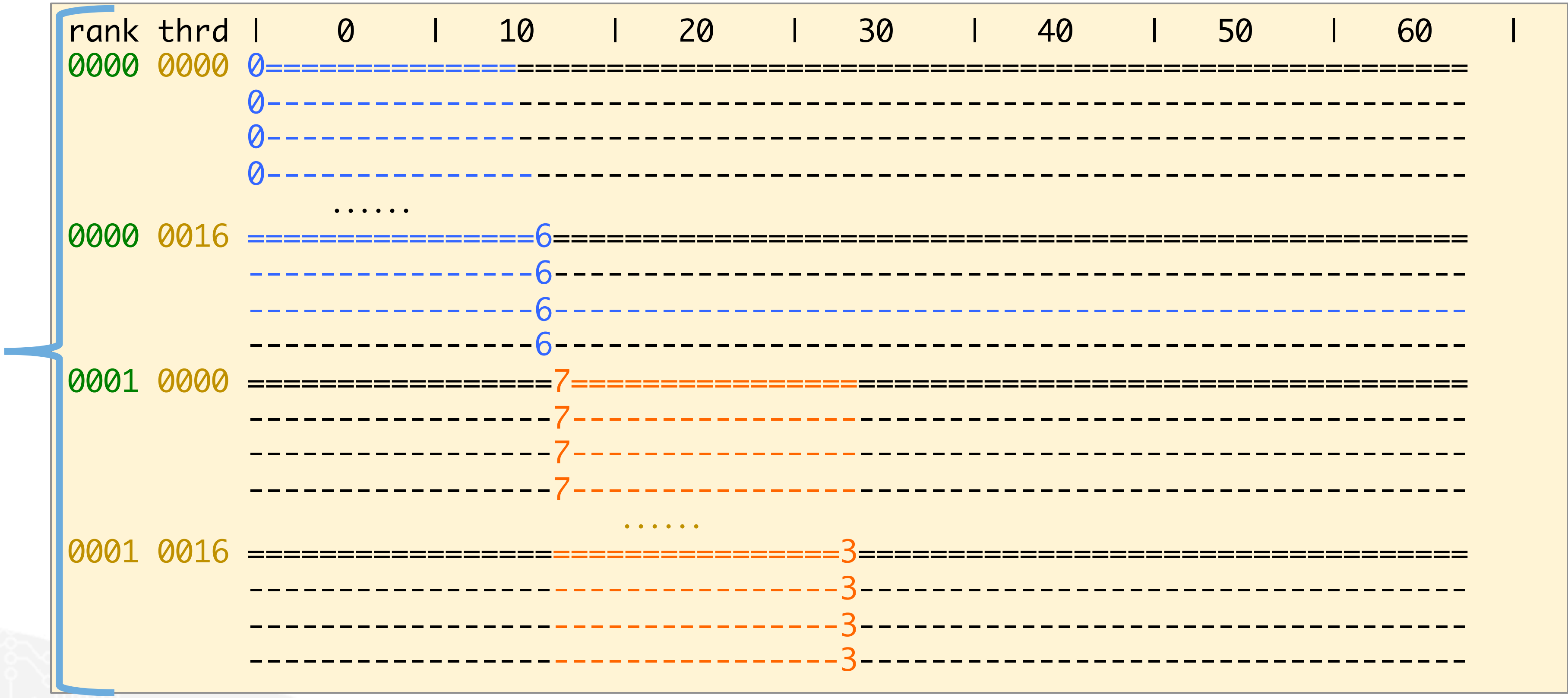
mpirun −np 4 amask_hybrid

First: Report MPI process affinities



```
rank |    0    |   10    |   20    |   30    |   40    |   50    |   60    |
0000 01234567890123456===================================================
     01234567890123456---------------------------------------------------
     01234567890123456---------------------------------------------------
     01234567890123456---------------------------------------------------
0001 =================78901234567890123==================================
     -----------------78901234567890123----------------------------------
     -----------------78901234567890123----------------------------------
     -----------------78901234567890123----------------------------------
                                          ....
0003 ===================================================12345678901234567
     ---------------------------------------------------12345678901234567
     ---------------------------------------------------12345678901234567
     ---------------------------------------------------12345678901234567
```

Next: Report Hybrid MPI process and OpenMP thread Affinities



```
rank thrd |    0    |   10    |   20    |   30    |   40    |   50    |   60    |
0000 0000 0===================================================================
          0-------------------------------------------------------------------
          0-------------------------------------------------------------------
          0-------------------------------------------------------------------
               ......
0000 0016 ================6===================================================
          ----------------6---------------------------------------------------
          ----------------6---------------------------------------------------
          ----------------6---------------------------------------------------
0001 0000 =================7==================================================
          -----------------7--------------------------------------------------
          -----------------7--------------------------------------------------
          -----------------7--------------------------------------------------
                               ......
0001 0016 =================================3==================================
          ---------------------------------3----------------------------------
          ---------------------------------3----------------------------------
          ---------------------------------3----------------------------------
```

amask

- Summary
  - Runs separate from app (or within code with API)
  - Reports single-character mask
  - Easy to determine proc-id, and layout for all processes/threads
  - Works for Multi-Node Environments
  - Creates Load for observing usage with *cpu_usage*, *htop*, etc.
  - Can Instrument code, has load utility, and timer.

Future Work

- cpu_usage: more details-- hover capability like google plots
- show_affinity: make dynamic GUI-like core_usage
- show_affinity: make GUI hardware aware (with colors)
- amask: color mask bits according to NUMA or socket id
- amask: different types of load (int/non-vec/vec, for HW monitors)
- amask: extract/display affinity from running processes

- Coordinate/Combine these three tools.

Thanks.

Questions?