Charm++ - Bug #2016
+autoProvision ignores numactl settings
10/26/2018 11:56 AM - Jim Phillips

<table>
<thead>
<tr>
<th>Status:</th>
<th>New</th>
<th>Start date:</th>
<th>10/26/2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee:</td>
<td>Evan Ramos</td>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Category:</td>
<td>Machine Layers</td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
<td>Spent time:</td>
<td>0.00 hour</td>
</tr>
</tbody>
</table>

Description

Runs on 64 threads when numactl is limited to 12:

policy: default
preferred node: current
physcpubind: 0 1 2 3 4 5 6 7 8 9 10 11
cpubind: 0 1
nodebind: 0 1
membind: 0 1 2 3 4 5 6 7
Charm++: standalone mode (not using charmrun)
Charm++ Running in Multicore mode: 64 threads (PEs)
Charm++ Using recursive bisection (scheme 3) for topology aware partitions
Converse/Charm++ Commit ID: v6.9.0-rc2-0-ge8d6e0c-namd-charm-6.9.0-build-2018-Oct-25-11430

Note that it does follow Comet's queueing system setup, and warns that there is something odd:

Intel(R) processor family information utility, Version 4.1 Update 3 Build 20131205
Copyright (C) 2005-2013 Intel Corporation. All rights reserved.

===== Processor composition =====
Processor name : Intel(R) Xeon(R) E5-2680 v3
Packages(sockets) : 1
Cores : 12
Processors(CPUs) : 12
Cores per package : 12
Threads per core : 1

===== Processor identification =====
Processor Thread Id. Core Id. Package Id.
0 0 0 0
1 0 1 0
2 0 2 0
3 0 3 0
4 0 4 0
5 0 5 0
6 0 8 0
7 0 9 0
8 0 10 0
9 0 11 0
10 0 12 0
11 0 13 0

===== Placement on packages =====
Package Id. Core Id. Processors
0 0,1,2,3,4,5,6,7,8,9,10,11,12,13 0,1,2,3,4,5,6,7,8,9,10,11

===== Cache sharing =====
Cache Size Processors
L1 32 KB no sharing
L2 256 KB no sharing
L3 30 MB (0,1,2,3,4,5,6,7,8,9,10,11)
policy: default
preferred node: current
physcpubind: 0 1 2 3 4 5 6 7 8 9 10 11
cpubeind: 0
nodebind: 0
membind: 0 1
/var/spool/slurmd/job19830854/slurm_script: line 14: nvidia-smi: command not found
Charm++: standalone mode (not using charmrun)
Charm+++ Running in Multicore mode: 12 threads (PEs)
Charm+++ Using recursive bisection (scheme 3) for topology aware partitions
Converse/Charm++ Commit ID: v6.9.0-rc2-0-ge8d6e0c-namd-charm-6.9.0-build-2018-Oct-25-11430
Warning: Randomization of virtual memory (ASLR) is turned on in the kernel, thread migration may not work! Run 'echo 0 > /proc/sys/kernel/randomize_va_space' as root to disable it, or try running with '+isomalloc_sync'.
CharmLB> Load balancer assumes all CPUs are same.
Charm+++ Warning: Internally-determined PU count does not match hwloc's result!