Support for Intel Knights Landing (KNL)

Morris Jette and Tim Wickberg
SchedMD LLC

Slurm User Group Meeting 2016

Outline

- KNL Overview
- KNL Scheduling Issues
- Node Features plugins
  - Cray system support
  - Generic clusters support

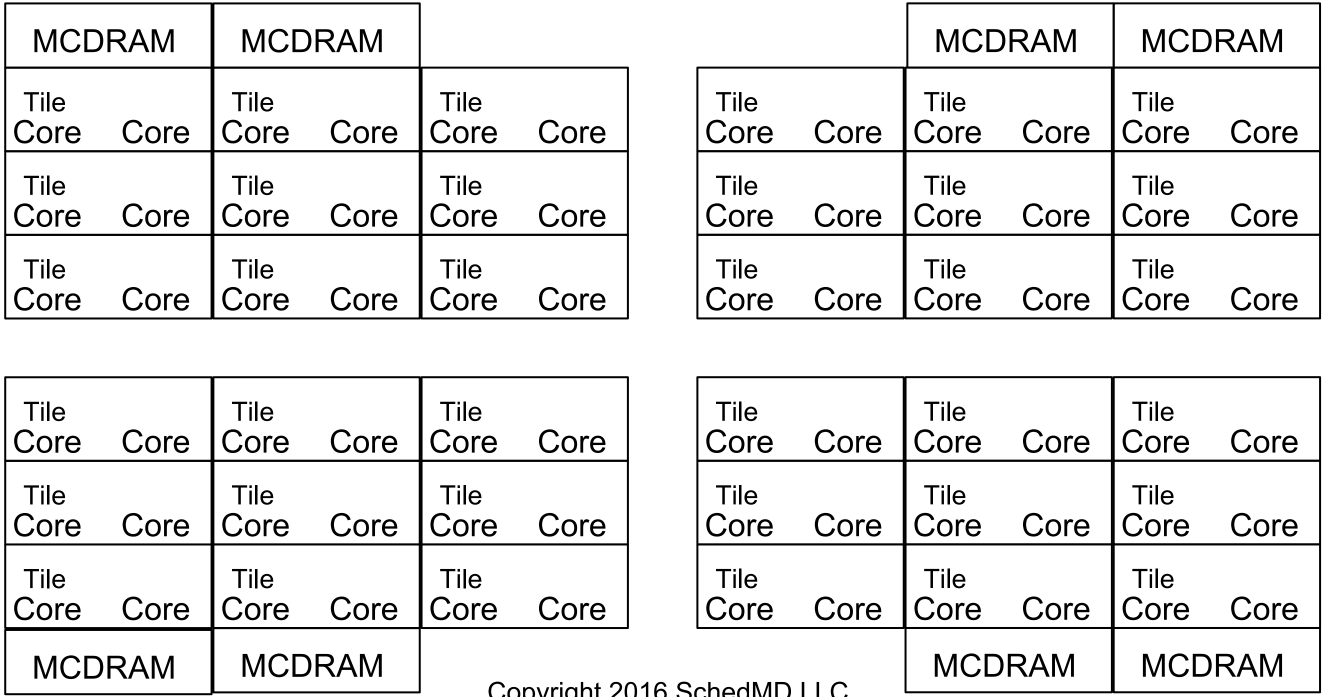
Intel Knights Landing (KNL) Overview

- Up to 72 Airmont (Atom) cores with four threads per core
  - Arranged in 2-D mesh interconnect
- Up to 384 GB of "far" DDR4 RAM
- 8 – 16 GB of stacked "near" 3D MCDRAM (Multi-Channel DRAM), a version of High Bandwidth Memory (HBM)
- Can be used as co-processor or self-boot (stand-alone processor)
  - Co-processor mode previously supported through gres/mic for KNC

KNL NUMA Modes

- The 2 dimensional mesh interconnect can be configured at boot time into one of five different modes
  - All-to-all (a2a): Uniform mesh interconnect
  - Hemisphere (hemi): Two virtual address spaces (one NUMA domain)
  - Quadrant (quad): Four virtual address spaces (one NUMA domain)
  - Sub-NUMA-2 (snc2): Two distinct NUMA domains
  - Sub-NUMA-4 (snc4): Four distinct NUMA domains

KNL SNC4 NUMA Mode



MCDRAM MCDRAM

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

MCDRAM MCDRAM

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

MCDRAM MCDRAM

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

Tile Core Core | Tile Core Core | Tile Core Core

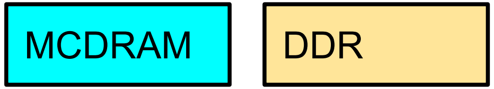
MCDRAM MCDRAM

KNL Memory Modes

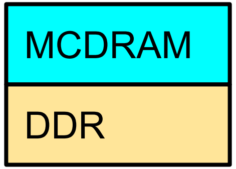
- The MCDRAM can be configured as cache, part of physical memory, or part cache + part memory
- The portion of MCDRAM configured as part of physical memory is known as High Bandwidth Memory (HBM)
- Reboot required to change memory mode

KNL Memory Modes

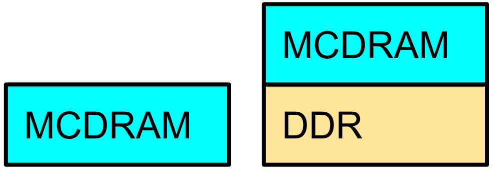
Cache Mode
MCDRAM entirely cache



Flat Mode
MCDRAM entirely memory



Hybrid Mode
Some of MCDRAM is cache,
Some of MCDRAM is memory



High Bandwidth Memory (HBM)

- Amount of available HBM can vary with MCDRAM mode
- HBM availability is managed as a Slurm Generic Resource (GRES) and can change at node boot time
- NOTE: Currently no mechanism in Slurm to ensure that a job does not consume more HBM than requested. This will be addressed in a future release (tentatively version 17.02)

Issues for Slurm

- Large core/thread count (72 cores, 288 threads)
- Changes to MCDRAM mode and HBM size at boot time
- Changes to NUMA mode and NUMA count at boot time
- Overhead of booting nodes before use
  - 5-7 minutes on a standalone system
  - … longer on a Cray KNL node

Node Features

- Used to establish node characteristics for scheduling purposes
- Split into two fields:
    - Available features: NUMA and MCDRAM modes which can be made available with a node reboot
    - Active features: Current NUMA and MCDRAM modes, possibly modified when computed node is booted

```
NodeName=nid00001
ActiveFeatures=quad,flat
AvailableFeatures=a2a,hemi,quad,snc2,snc4,cache,split,flat
```

Node Features: Scheduling

- User specifies required mode on job command line
  - Only AND operation supported, no OR, XOR, counts, etc.
- Job will be allocated nodes already in desired mode if possible
- Nodes will be rebooted only if needed
  - Boot time can be tens of minutes, avoid if possible

```
sbatch -C a2a,flat -n 72 -N1 --hint=nomultithread my.bash
```

Node Features: Scheduling

- The job is billed for all resources from the time of allocation
  - Boot time is charged against job in fairshare and sacct
    - Looking at splitting the CF and R times apart in future
  - Not counted against the TimeLimit for the job
- Nodes can only be rebooted it has no active jobs
  - Could prove problematic if resource allocations not at node level (e.g. different cores allocated to different jobs)

Node Features Plugin

- Provides mechanism to get and modify a node's MCDRAM and NUMA configuration plus boot the node
- Configuration file with administrative options
- Two plugins available
  - knl_cray for Cray systems
  - knl_generic for generic clusters

knl_cray Plugin

- Available today (version 16.05.0+)
- Cray's *capmc* and *cnselect* commands used to:
  - Read current MCDRAM and NUMA mode
  - Change MCDRAM and NUMA mode
  - Reboot nodes
  - Test node status
- All operations performed on head node

knl_cray Plugin

- Makes use of Slurm infrastructure to suspend idle nodes and return them to service as needed
  - Capmc_suspend and capmc_resume programs in the contribs directory should be installed and configured in slurm.conf as SuspendProgram and ResumeProgram
  - Configure SuspendTime to large value if suspending of idle nodes is not desired

knl_cray Plugin

- If node mode change or boot fails, the *capmc* command currently does not identify the failing node
  - The job allocated those nodes will be requeued and held
  - Nodes previously allocated to the job can be used in subsequent resource allocations until the bad node(s) can be identified

knl_generic Plugin

- Available October 2016 (version 16.05.6)
  - Code written and being tested
- Intel’s *syscfg* command used to
  - Read current MCDRAM and NUMA mode
  - Change MCDRAM and NUMA mode
- Linux *reboot* command used to
  - Reboot nodes
- All operations performed directly on compute nodes

knl_generic Plugin

- If node mode change or boot fails
  - The bad node(s) will be set DOWN
  - The job allocated those nodes will be requeued and scheduled when possible

knl.conf Configuration File

- Who is allowed to reboot nodes
- Available MCDRAM and NUMA modes
  - Could be subset of those supported by the processor
- Default MCDRAM and NUMA modes
- Path to programs used to get/set mode information
- Timeouts for called programs
- Different parameters for Cray and generic systems

Caveats

- Slurm currently only supports homogeneous NUMA
  - 68-core KNL in SNC4 or Quadrant mode not supported
    - Results in unbalanced NUMA domains of [16, 16, 18, 18] cores
    - Scheduler requires all domains to match
- Recommend CoreSpecCount to minimize OS jitter
  - Linux kernel can keep 2-4 cores 100% busy under load

More Information Online

https://slurm.schedmd.com/intel_knl.html

https://slurm.schedmd.com/knl.conf.html

Questions?