Node Resource Management in Next-Generation Systems

Edgar A León
21 May 2021

Source: https://comicvine.gamespot.com/radar/4005-83379/

RADR: Workshop on Resource Arbitration for Dynamic Runtimes

LLNL-PRES-816236. LLNL-VIDEO-822794.
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344.
Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

Talking about superheroes… Thank you!

- Balazs Gerofi
- Brice Goglin
- Samuel Gutierrez
- Matthieu Hautreux
- Julien Jaeger
- Guillaume Mercier
- Rolf Riesen
- Masamichi Takagi

RIKEN

cea

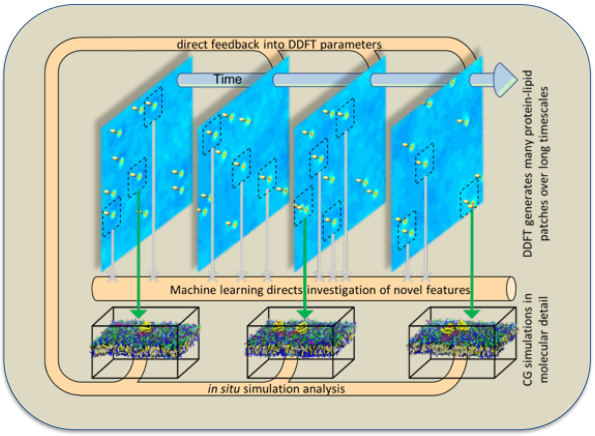
Los Alamos NATIONAL LABORATORY

intel

Inria

Increasingly complex workflows are pushing the limits of HPC software environments

- Multi-physics simulations
  - National security applications
- Data science workloads
  - TensorFlow
  - PyTorch
  - LBANN
- Multi-kernel applications
  - Weather prediction models
- Cognitive simulations
  - Conventional HPC + Deep-Learning



A Massively Parallel Infrastructure for Adaptive Multiscale Simulations.
Di Natale et al., SC 2019

Heterogeneous supercomputing systems pose challenges to application and library developers

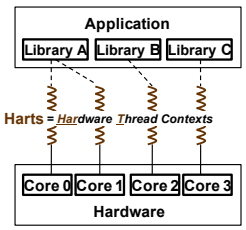
- Users must utilize a combination of programming models and runtimes
- Components assume dedicated resources
  - Coordination is left to application and library developers

Heterogeneous Systems	Growing Gap	Application Needs
Accelerators		Portability
Latency opt processors	MPI OpenMP, Pthreads US DOE RAJA, Kokkos CUDA, HIP, OpenMP 4+	Productivity
Heterogeneous memory		Performance

Poor or no coordination of resources among node-local components

While not new, this problem is exacerbated by workflow complexity and system complexity

- Promising research in this area
  - Lithe, QUO
  - Argo, Hobbes, multi-kernels
  - Resource and workflow managers



H. Pan's PhD Dissertation, MIT 2010

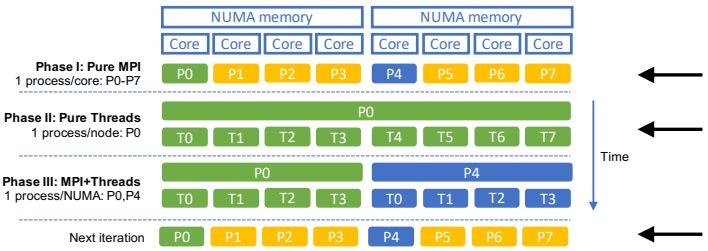
- Yet, problem is getting worse

This work:

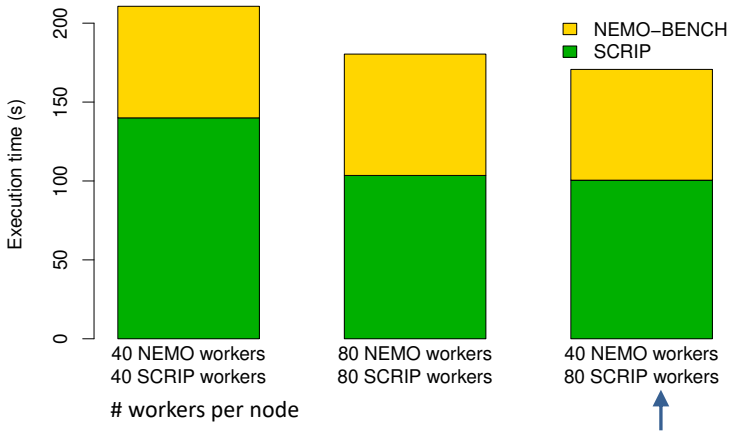
- Understand requirements of current and emerging applications
- Group challenges into a handful of themes we can study
- Propose a strawman solution for application composition

Climate modeling applications integrate multiple kernels with different runtime configurations

- Multiple domains and models
  - Ocean physics
  - Atmospheric physics
  - Biogeochemistry
- Model-specific kernels are developed independently
  - Need to exchange data
  - May use different runtimes
  - May have different optimal points



Thread heterogeneity from different models in one application is difficult to program and impacts performance



- OASIS from CERFACS
- NEMO-BENCH
  - Oceanic circulation model benchmark
  - MPI parallelism
- SCRIP
  - Interpolation library
  - OpenMP parallelism

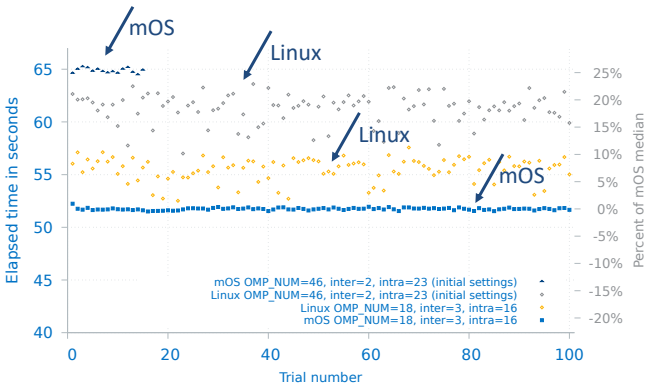
Each compute node: 40 cores, 2 HT/core

Coordinating multiple components, each with a different type and number of workers, is challenging and error-prone

- Inertial Confinement Fusion
  - LLNL's National Ignition Facility
  - Livermore Big Artificial Neural Network
- I/O
  - C++ threads on CPU
- Compute
  - OpenMP threads on CPU
  - GPU kernels
- Communication
  - Aluminum Pthreads
  - NCCL Pthreads

Conflicting directives from different parts of the software stack may hinder performance and the ability to optimize

- Deep learning in Cancer problems
  - CANDLE benchmark
    Pilot 3 data set from ECP
  - TensorFlow + Intel MKL-DNN
- TensorFlow
  - Two thread pools to stage work
  - User controls size but not placement
- Intel MKL-DNN
  - OpenMP threads
  - User controls size and placement



Compute node with 48 cores

This work:

- Understand requirements of current and emerging applications
- Group challenges into a handful of themes we can study
- Propose a strawman solution for application composition

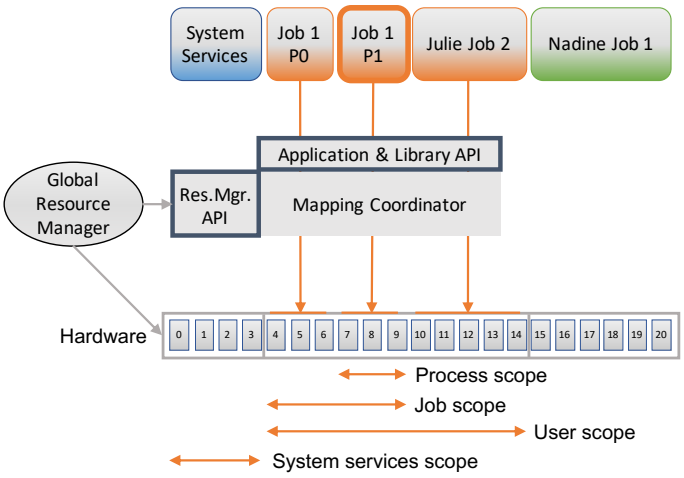
Created 4 themes to capture application requirements and simplify their study

	CANDLE	GeoFEM	NWChem	NEMO BENCH	Hydro	ICF LBANN	SCALE-LETKF
Multiple types of workers	■					■	
Dynamic compute workers	■	■	■	■	■	■	■
Dynamic utility workers		■	■			■	
Remapping of tasks/threads				■	■		■
Multiple applications				■			■
MPI	■	■	■	■	■	■	■
OpenMP	■	■	■	■	■	■	■
POSIX threads						■	
NVIDIA CUDA						■	
C++ threads	■					■	

This work:

- Understand requirements of current and emerging applications
- Group challenges into a handful of themes we can study
- Propose a strawman solution for application composition

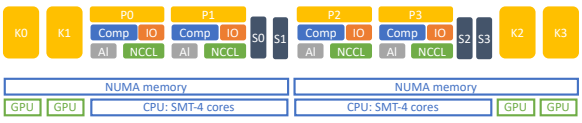
A *Mapping Coordinator* can provide the functionality needed to meet application demands



- Provides two-prong interface
  - Low-level / high-level functions
- Reconfigures worker types dynamically
- Manages node-local HW
  - Affinity and binding
  - Query availability
  - Request & release
  - Arbitrate access
- Coordinates with the global resource manager

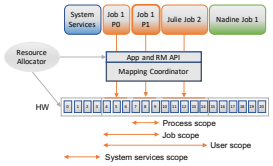
Enabling application composition will be paramount for emerging science applications on tomorrow's systems

- Components assume dedicated resources
  - Coordination left to app/library developers
- Problem not new, but is getting worse
  - Application workflows are more complex
  - Systems are more heterogeneous



León et al. Application-Driven Requirements for Node Resource Management in Next-Generation Systems. ROSS 2020. IEEE.

- Identified challenges based on real applications
  - Multiple, uncoordinated types of threads
  - Dynamic work by auxiliary libraries
  - Rebalance and remap of workers
  - Multiple applications working together
- Proposed a Mapping Coordinator strawman to mitigate challenges




Lawrence Livermore
National Laboratory
Disclaimer
This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.