OAK RIDGE National Laboratory

Mixed Feelings about Mixed Precision?

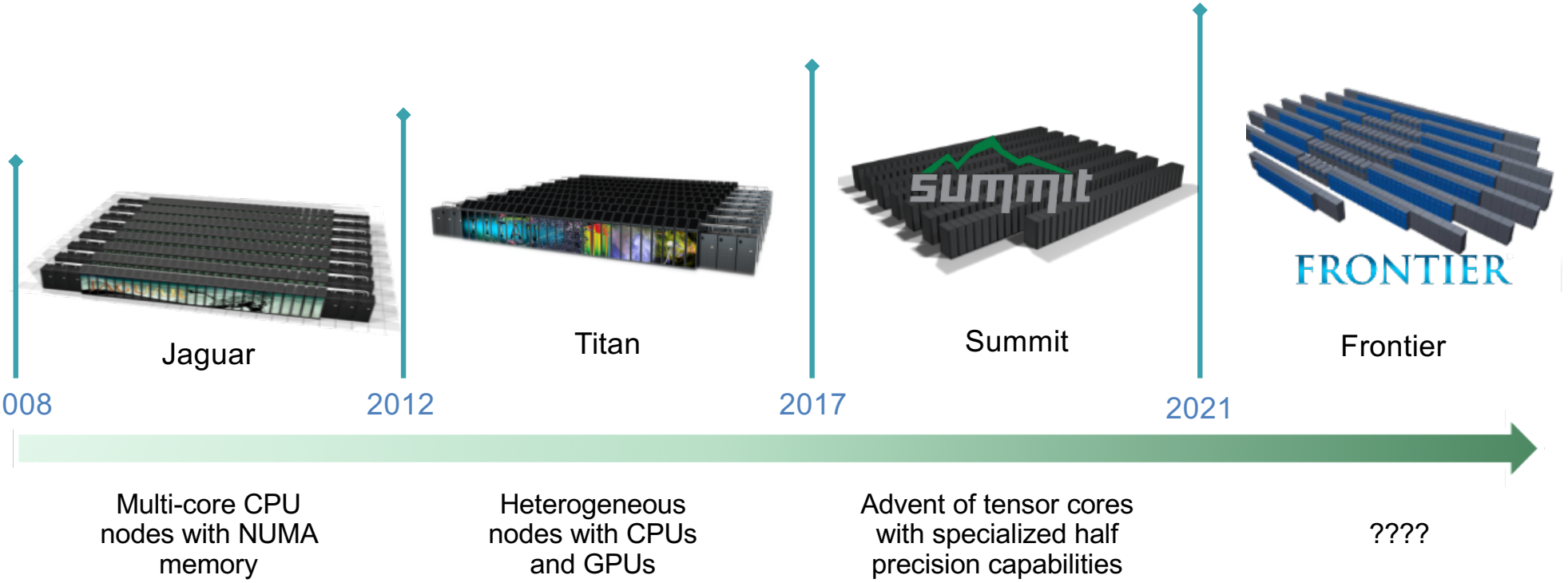
Judy Hill
Scientific Computing Group Leader, Center for Computational Sciences

Stuart Slattery
Computational Scientist, Computational Sciences and Engineering

August 27, 2019
Smoky Mountains Computational Science and Engineering Conference
Kingsport, TN

U.S. DEPARTMENT OF ENERGY

Trend of hardware becoming more specialized …



Developers of application software and their underlying technology have historically adapted to these trends … but many have mixed feelings about mixed precision

Mixed precision is not new …

- In the 1970's, seminal work by Chris Paige showed the effects of rounding errors in the Lanczos algorithm can be described by rigorous theory
  - C. C. Paige (1971), The computation of eigenvalues and eigenvectors of very largesparse matrices, PhD thesis, University of London.
- Subsequent work has demonstrated that mixed precision methods for iterative schemes converge
  - e.g. R. Strzodka and D. Goddeke, Mixed Precision Methods for Convergent Iterative Schemes, In Proceedings of the 2006 Workshop on Edge Computing Using New Commodity Architectures, May 2006.
- Today, performance-driven tuning is proposed to identify regions of code where lower precision may be appropriate
  - e.g. Ignacio Laguna, Paul C. Wood, Ranvijay Singh, and Saurabh Bagchi. GPUMixer: Performance-Driven Floating-Point Tuning for GPU Scientific Applications. Proceedings of ISC 2019.

Speakers for this session

Enabling Technologies

Applications

Jack Dongarra, University of Tennessee
"Using Mixed Precision in Numerical Computation"

Daniel Osei-Kuffuor, Lawrence Livermore National Lab
"Variable Precision Computing for Scientific Applications"

Kate Clark, NVIDIA
"Effective use of Mixed Precision for HPC"

Thomas Maier, Oak Ridge National Lab
"Mixed Precision Sampling of Quantum States of Matter"

Vikram Gavini, University of Michigan
"Fast, Scalable and Accurate Finite-Element Based Ab Initio Calculations Using Mixed Precision Computing"

Questions to consider during this session

- Mixed precision methods (single vs double) have been around, but are there opportunities in applications where even lower precision makes sense?
  - Am I guaranteed the stability, accuracy and convergence properties using lower precision? **[Am I going to get the right answer?]**
  - What memory and performance improvements can I expect when using lower precision? **[Is this worth my time?]**
  - What implementation challenges exist for application and enabling technologies developers? **[How long will it take me to put this into my code?]**

Is now the time to "drop everything" to investigate algorithmic changes?