SC20

Everywhere we are | more than hpc.

Panel: TBAA: Task-Based Algorithms and Applications

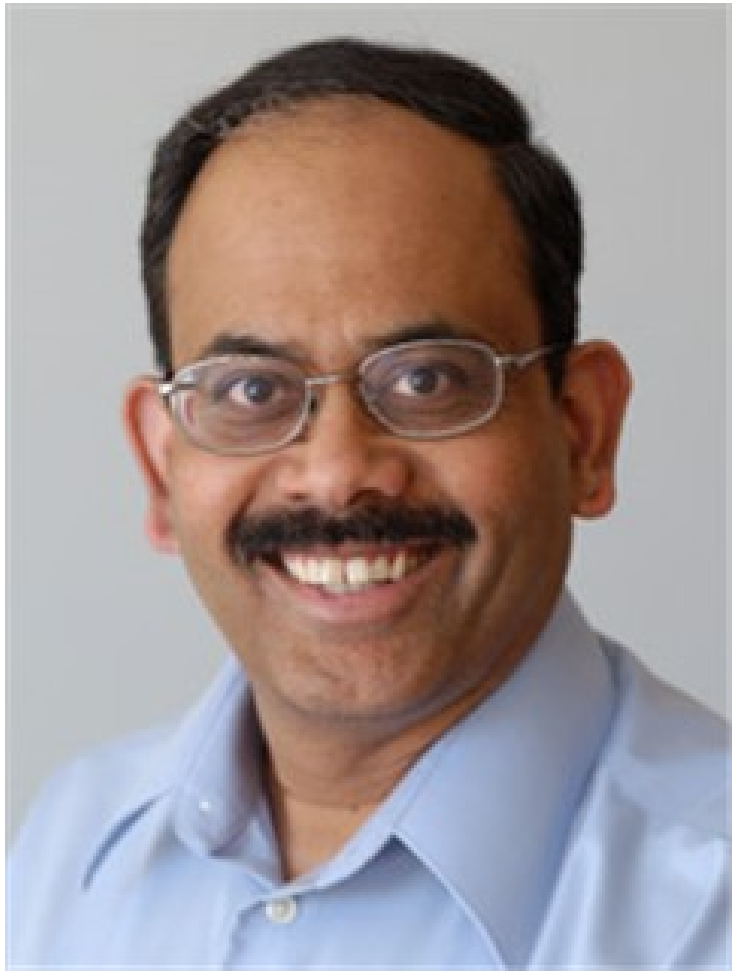
Wednesday, 18 November 2020 • 3:00 pm – 4:30 pm

Moderator: Patrick Diehl, Louisiana State University, Center for Computation and Technology

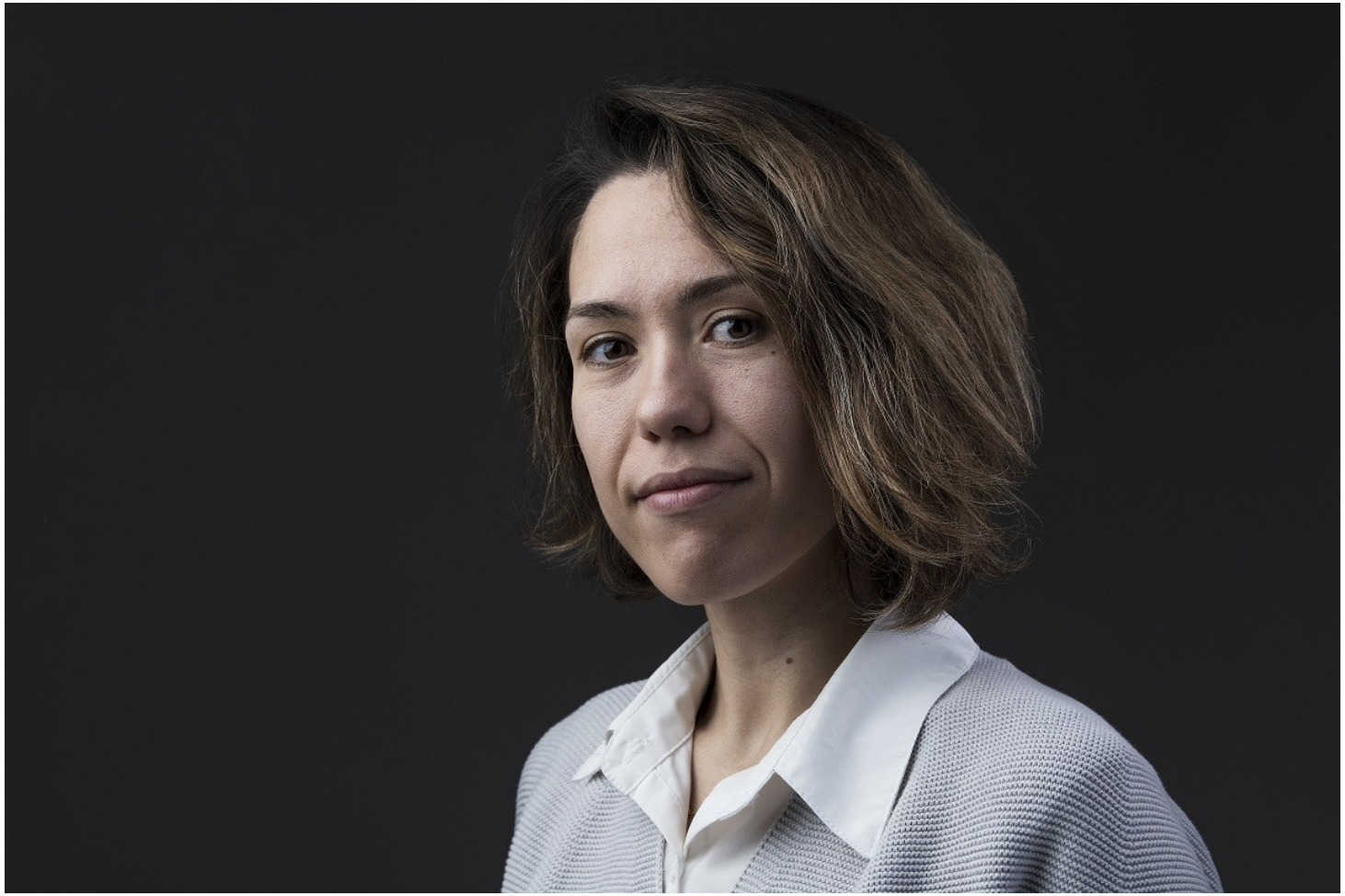
Outline

1. Introduction of panelists
2. Introduction of Asynchronous many-task systems
3. State of the art:
   - Charm++
   - The Julia Programming Language
   - The C++ standard library for parallelism and concurrency
4. Panel's Chosen Questions
5. Q & A from audience (live)

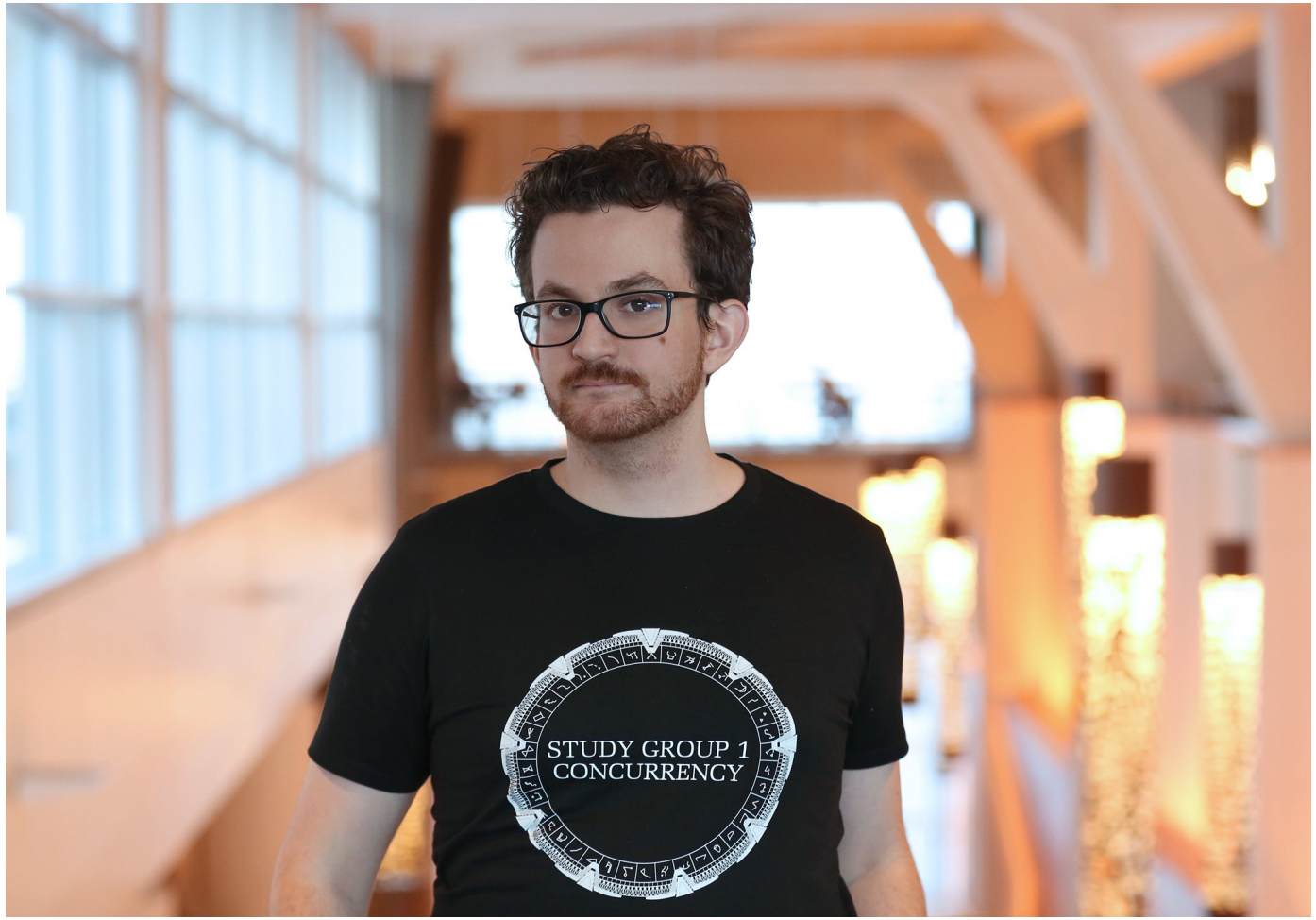
Panelist I: Laxmikant V. Kale, University of Illinois



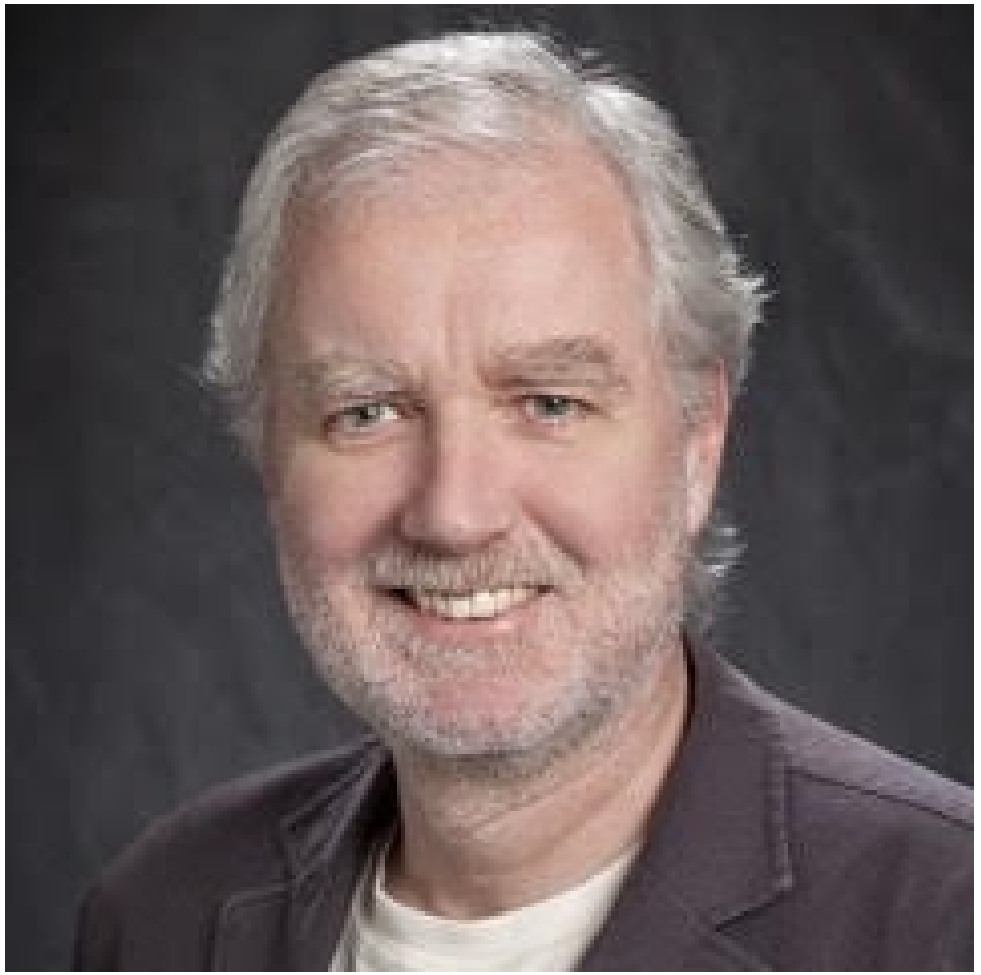
Panelist II: Irina P. Demeshko, Los Alamos National Laboratory



Panelist III: Bryce Adelstein Lelbach, NVIDIA

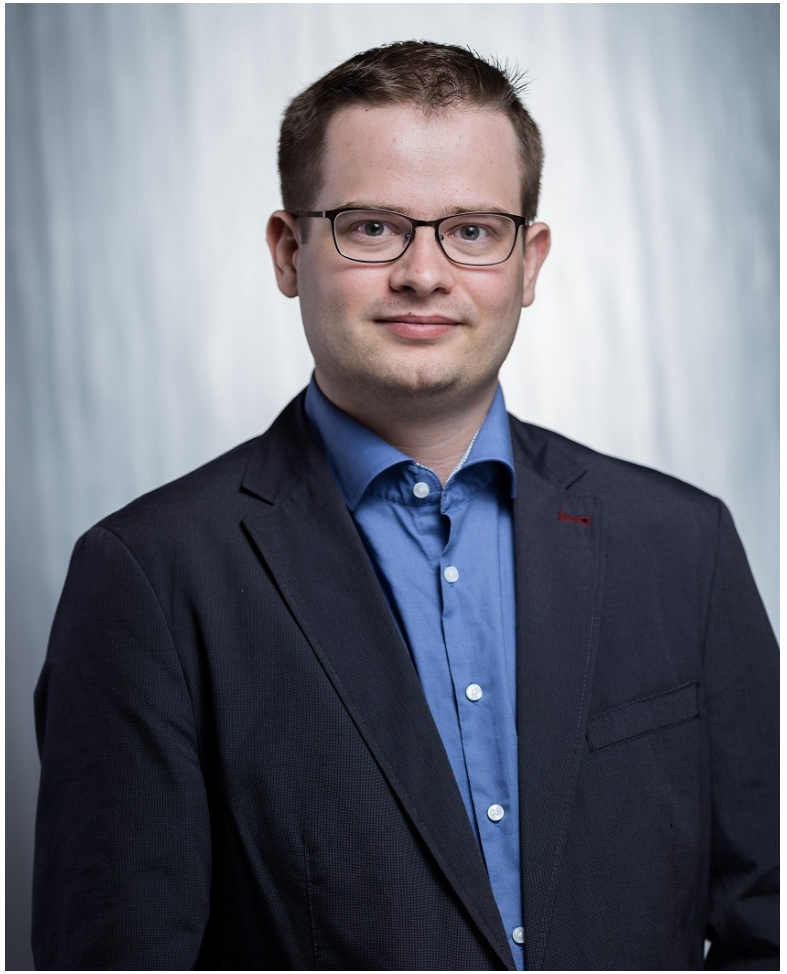


Panelist IIII: Hartmut Kaiser, Louisiana State University, Center for Computation & Technology

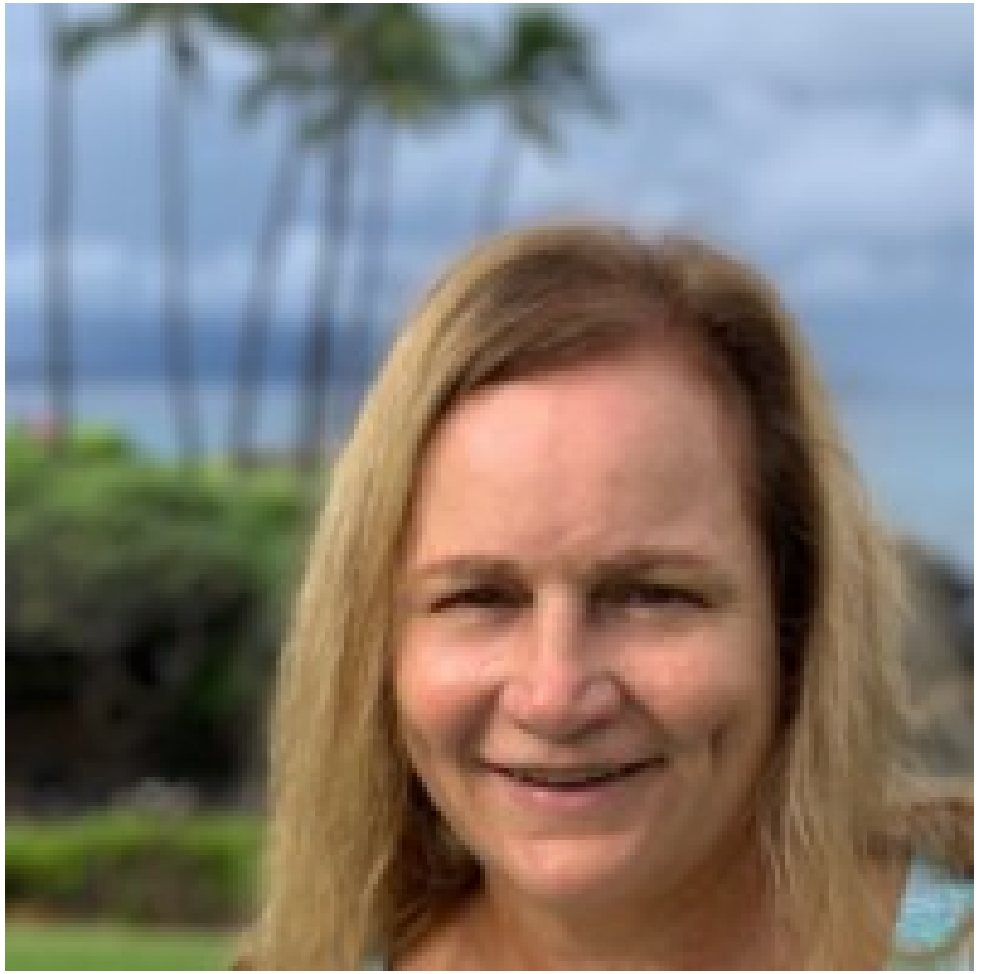


Panelist V: Zahra Khatami, Oracle

Panelist VI: Keno Fischer, Julia Computing Inc



Panelist VII: Alice Koniges, University of Hawaii



Panel's Chosen Questions I

- For what class of applications is an AMT paradigm the best solution for achieving the scalable execution?

Panel's Chosen Questions II

- How does the C++ language standard impact efficiency and complexity of programming AMT systems?

Panel's Chosen Questions III

- What are the differences between HPX, Julia, and Charm++ AMT paradigms and how can these differences affect the parallel performances?

Panel's Chosen Questions IIII

- Have these HPX, Julia, and Charm++ AMT paradigms performances been evaluated against scientific applications, AI, or any other frameworks? If so, how did those frameworks get benefit using these AMT models compared to the traditional runtime systems?

Panel's Chosen Questions V

- How does the supercomputing community benefit from using AMT on modern and future supercomputers?

Panel's Chosen Questions VI

- What hardware features are required in the next-generation processors to support AMT?