Challenges and Solutions for Peta- and Exa-Sacle Programming

Tasuku Hiraishi
Academic Center for Computing and Media Studies, Kyoto University

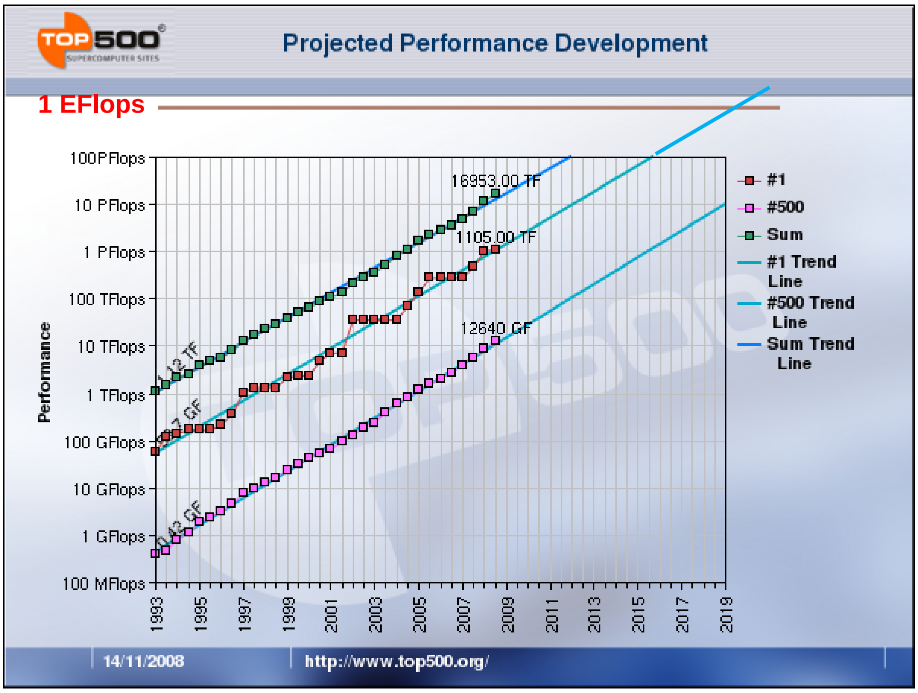
- (0) When Exa-scale system will come ?
  - Year 201X ?
  - Never come :-)

  (1) Issues on Peta- and Exa-scale programming:
  What issues are solved or are beeing solved ?
  What issues will be remained at the end ?
  (2) Programming style:
  What kind of programming style remains ?
  - Explisit message passing (MPI or other)
  - Shared memory & multi-thread (OpenMP or other)
  - Brand new one (I will make it!)

  Does the user have to describe Hybrid code ?
  (3) Your challenge:
  What is your most difficult challenge for Peta- and Exa-scale software ?
  - Application software
  - System software

  (4) Your free opinion on Peta- and Exa-scale application/system software.

When Exa-scale system will come ?



Issues on Peta- and Exa-scale programming (1)

- Peta- and Exa-scale systems will be realized by
  - O(10^2)-O(10^3) nodes
    - distributed memory
  - O(10^2)-O(10^3) cores per node
    - hierarchical shared memory?
  - heterogeneous environment? (GPGPU?)

→needs more complicated programming to obtain maximum performance

- OpenMP, MPI, (and xScalableMP) will still remains

Issues on Peta- and Exa-scale programming (2)

- Clusters and multicores will become more and more common
- → More and more (non-numerical) users will need parallel computing only for *acceptable* performance
- → We will need highly productive programming environment (language and/or library)
  - *reasonable* performance at low cost
- MapReduce [Google]
- X10 [IBM], Fortress [Sun], Cilk++, etc.?
- High-level programming languages (esp. functional languages) for parallel (distributed) environment

Conclusion (my challenge)

I'd like to make Lisp common among HPC users!