Revolutionary Approaches
Initial Discussions

William Gropp, UIUC

Thomas Sterling, IU

Engaging Application Developers

- Education
- Early adopter platform availability
- Long term horizon for new methods
- Tools for data decomposition
  - Didn't happen
  - Too problem specific
- Computational scientists in need
  - How do you make the case?

Where can revolutionary approaches succeed

- Applications that have given up on extreme scale
- Current applications exhibiting poor scaling
- Complementing other non-revolutionary approaches
- E.g., Pointer arithmetic
- An economic argument for general acceptance
- Note: reliability factors need to be better quantified
- Quantify the threat to Exascale.
  - Move from qualitative and intuitive to quantitative and repeatable

Trailblazing

- Need a few good people
- Need infrastructure with longevity
- Give them a place where they can succeed
- Learn from what has failed in the past
- We need to start, somehow
- Undertake point-designs
  - E.g. co-design
- Experience possibilities and weed out or add as appropriate

Practical Constraints

- Major issue is cost
  - Capital or acquisition cost
  - Operating cost
  - Memory capacity an example
- Revolution can be:
  - Much cheaper
  - Much easier
  - Making it possible