ASCR Programming Challenges Workshop

William Harrod
Sonia R. Sachs
July 27-29, 2011
Welcome and Goals

- Welcome
- Workshop Goals:
  - Define objective criteria for assessing programming models and language features that enable effective use of diverse Exascale architectures for important science applications.
  - Prioritize challenges for programming models, languages, compilers and runtime systems for Exascale
  - Prioritize options for
    - evolutionary path,
    - revolutionary path and
    - bridging the gap between evolutionary and revolutionary paths
  - Create a roadmap, with options, timeline, and rough cost estimates for programming Exascale systems that are responsive to the needs of applications and to future architectural constraints
State-of-the-art Session I

• **Presentations** on advanced programming models and languages, *describing and comparing* capabilities and advantages and disadvantages of approaches. 20 minutes + 5 minutes for questions.

• **Focused Parallel Panel discussions**
  – Develop objective criteria to assess programming models considering various models of computation primitives:
    • Communication and Synchronization Primitives Panel
    • Scheduling Primitives Panel
    • Partitioning and Placement Primitives Panel

• **Session I General Panel**
Explaining Focused Panels for Session I

• These primitives apply at all levels of abstraction:
  ➔ algorithm ➔ execution model ➔ programming model ➔ language ➔ machine model

• We are focusing today on programming models

• We are here to explore how these primitives are defined in Exascale environments
Explaining Focused Panels for Session I

- **Communication:**
  - describes how work and data are passed from one parallel task to another (broadcast, multicast, point-to-point, near neighbor, tree, etc.)

- **Synchronization:**
  - describes the control and data mechanisms for coordinating parallel operations (producer-consumer, barrier, locks)

- **Partitioning:**
  - describes how work and data are split between different physical resources (what to run as threads, what is the grain size, division of work...)

- **Placement:**
  - describes the location of first class objects throughout the system (where to run, where to place the data...)

- **Scheduling:**
  - describes the ordering of work (when to run, static or dynamic, user-level or system-level... )
A Few Words about Exascale Challenges

• **Asynchrony** will be needed at all levels in Exascale computing:
  ➔ Algorithms ➔ execution models ➔ programming models
  ➔ languages ➔ machine models.

• The **paradigm shift** from bulk-synchronous computing to asynchronous computing appears unsettling and chaotic to many.
  – Not to worry:
    • From a theoretical, formal methods view point, we have shown\(^1\) that one can model asynchrony with a synchronous model.

• On the other hand, **this may only apply if the abstractions that we use** in the new asynchronous, massively parallel environment are **good enough** so that the theory applies...

---

Concurrent programming is difficult\(^1\)

Our physical world is highly concurrent, so why is concurrent programming difficult?

- Have we chosen incorrect programming abstractions?
- Are threads an example of such incorrect abstractions?
  - “achieving reliability and predictability using threads is essentially impossible for many applications.\(^2\)”
- Is message passing another example of incorrect abstraction?
  - “Message passing can be made as non-deterministic and difficult to understand as threads.\(^2\)”

A Few Words about Exascale Challenges

• Do we have examples of good abstractions?
  – In embedded systems, **actor-oriented programming**\(^1\) used **in the context of several models of computation** (Kahn Process Networks, Synchronous/Reactive, and Discrete Events) very naturally expresses concurrency.
  – We hope that at this workshop we will explore many abstractions to deal with asynchrony.

---

Workshop Organization

- Our Special Thanks to Bob Lucas for hosting this workshop
- Our Thanks to:
  - The Workshop Committee
    - Saman Amarasinghe (MIT),
    - Mary Hall (U. Utah),
    - Pat McCormick (LANL),
    - Richard Murphy (Sandia),
    - Keshav Pingali (U. Texas-Austin),
    - Dan Quinlan (LLNL),
    - Vivek Sarkar (Rice),
    - John Shalf (LBNL).
  - The Advisory Committee:
    - Bob Lucas (USC/ISI)
    - Kathy Yelick (LBNL/UCB)
  - Participants who contributed panel questions
  - ASCR Website team: Tom Monahan and Ginger Kirkendall
  - Support from Sandia, ISI, and ORISE (Larry Godinez, Dolores Cadena, Jeannie Robinson, and Deneise Terry)