Understanding Temporal Dynamics of Parallel Codes

John Mellor-Crummey
Department of Computer Science
Rice University
johnmc@cs.rice.edu

http://hpctoolkit.org
Acknowledgments

• Staff
  — Laksono Adhianto, Nathan Tallent

• High School Summer Interns
  — Sinchan Banerjee, Michael Franco, Chas Jhin, Reed Landrum

• SciDAC project support
  — Center for Scalable Application Development Software
    – Cooperative agreement number DE-FC02-07ER25800
Profiling compresses out the temporal dimension
- result: transient behavior, e.g. serialization, is invisible in profiles
What can we do? Trace call path samples
Key Steps

• **Measurement**
  – collect a CCT per process and/or thread using sampling
  – collect a trace of \(<\text{node id}, \text{timestamp}>\) pairs
    • note: arrange for one node-id per procedure

• **Post processing**
  – combine the CCTs into a canonical CCT
  – renumber all traces for consistency using canonical CCT

• **Visualization**
  – assign colors to procedures based on labeled nodes in CCT
  – only read in trace records you need to color the display’s pixels
  – views
    • space-time view: call stack trace for all threads at depth \(d\)
    • depth view: call stack trace for an individual thread at all levels
Flash White Dwarf Collapse on 256 Cores

Full execution at call stack depth 2
Flash White Dwarf Collapse on 256 Cores

Full execution at call stack depth 5
Flash White Dwarf Collapse on 256 Cores

Execution detail at call stack depth 5