Component Integration and Optimization

LACSI Priorities and Strategies Workshop 2005

Ken Kennedy
Rice University

http://lacsi.rice.edu/meetings/internal/slides_feb05/components.pdf
Plan for FY 05

• Refocus on Marmot as Component Challenge Problem
  — Interactions at Monterrey Workshop and a follow-up meeting at LANL (June 2004)
  — Abstract Mesh data structure to increase flexibility
  — Develop plan for activity by Q4 FY04

• Supporting Technologies for Component Integration
  — Transformation systems to eliminate overheads due to abstraction
  — Component integration systems to automate specialization
    - Key problem: integration of data structure components with functional components

• Retargetable High Performance Components
  — Pretuning arbitrary apps to new architectures
Component Integration

• Supporting Technologies for Component Integration
  — Transformation systems to eliminate overheads due to abstraction
  — Component integration systems to automate specialization
    - Key problem: integration of data structure components with functional components

• Continue Collaborations with Marmot Project
  — Pursue directions in the draft collaboration plan (next slide)
  — Application of object-oriented optimization strategies (from JaMake)

• New LANL Contact from Traditional Code Projects

• Challenge Application
  — Export-restricted version of hydro+radiation transport
  — Representative of “traditional” code projects
New Directions

• Specialization Strategies
  – Specialized handling of multiple materials in cells
  – Compiler-based specialization to sparse data structures
  – Combined telescoping languages and dynamic code selection
    – Optimization by limited computation reorganization

• Tools for Preoptimization of Libraries
  – Pre-specialization of library codes to expected calling contexts
  – Potential source of components: Trillinos

• Mining of Traditional Applications
  – Construction of libraries for inclusion in domain languages

• Rapid Prototyping Support
  – Compilation of scripting languages (Python, Matlab) to Fortran/C
Automatic Component Tuning

- Participants: Four Groups within LACSI
  - Tennessee: Jack Dongarra
    - Collaboration with LLNL ROSE Group (Dan Quinlan, Qing Yi)
  - Rice: Ken Kennedy and John Mellor Crummey
    - Students Apan Qasem and Yuan Zhao
  - Rice: Keith Cooper, Devika Subramanian, and Linda Torczon
    - Students Todd Waterman and Alex Grosul
  - Univ of Houston: Lennart Johnsson
    - Students Ayaz Ali, Purvi Shah, Haiyan Teng
Automatic Tuning Plan

• Retargetable High Performance Components
  — Pretuning components to new architectures
    - Arbitrary components: Heuristic search strategies
    - Structural approach: refactor the component into codelets
  — Fault Tolerant Algorithms

• Connection to LANL
  — Point of contact within LANL from “traditional” code projects
  — Release of kernels from code projects (export restricted?)

• Autotuning Challenge
  — Four teams within LACSI will apply techniques to the LANL kernels

• Longer Term
  — Application to component integration challenge application
Planned Workshops

• **Automatic Tuning**
  - **LACSI Groups**
    - Rice, UH, Tennessee
    - Cornell-UIUC (Pingali and Padua)
    - USC ISI (Mary Hall)

• **Parallel Scripting Languages**
  - **Through DARPA HPCS**
  - **Matlab Groups**
    - MIT (Kepner)
    - Tennessee (Dongarra)
    - Rice (Kennedy, Mellor-Crummey, Fowler)
    - OSC-Indiana-PNL (Ahalt, Sadayappan, Chauhan)