SAND2011-1191C

Miniapplications: Vehicles for Co-design

Michael A. Heroux

Scalable Algorithms Department

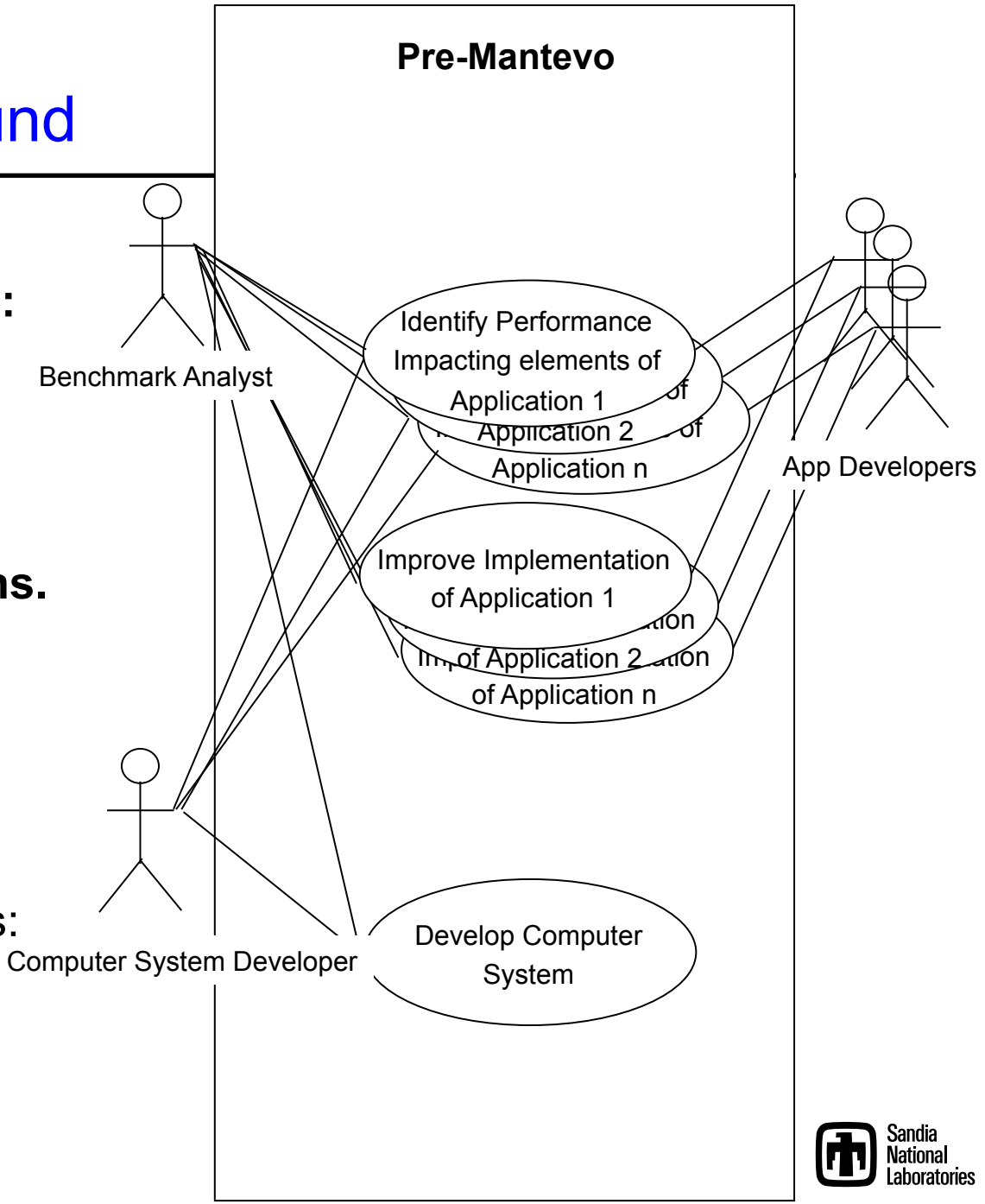
Recent Collaborators: Brian Barrett, Richard Barrett, Erik Boman, Ron Brightwell, Paul Crozier, Doug Doerfler, Carter Edwards, Kevin Pedretti, Heidi Thornquist, Alan Williams, Michael Wolf

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Background

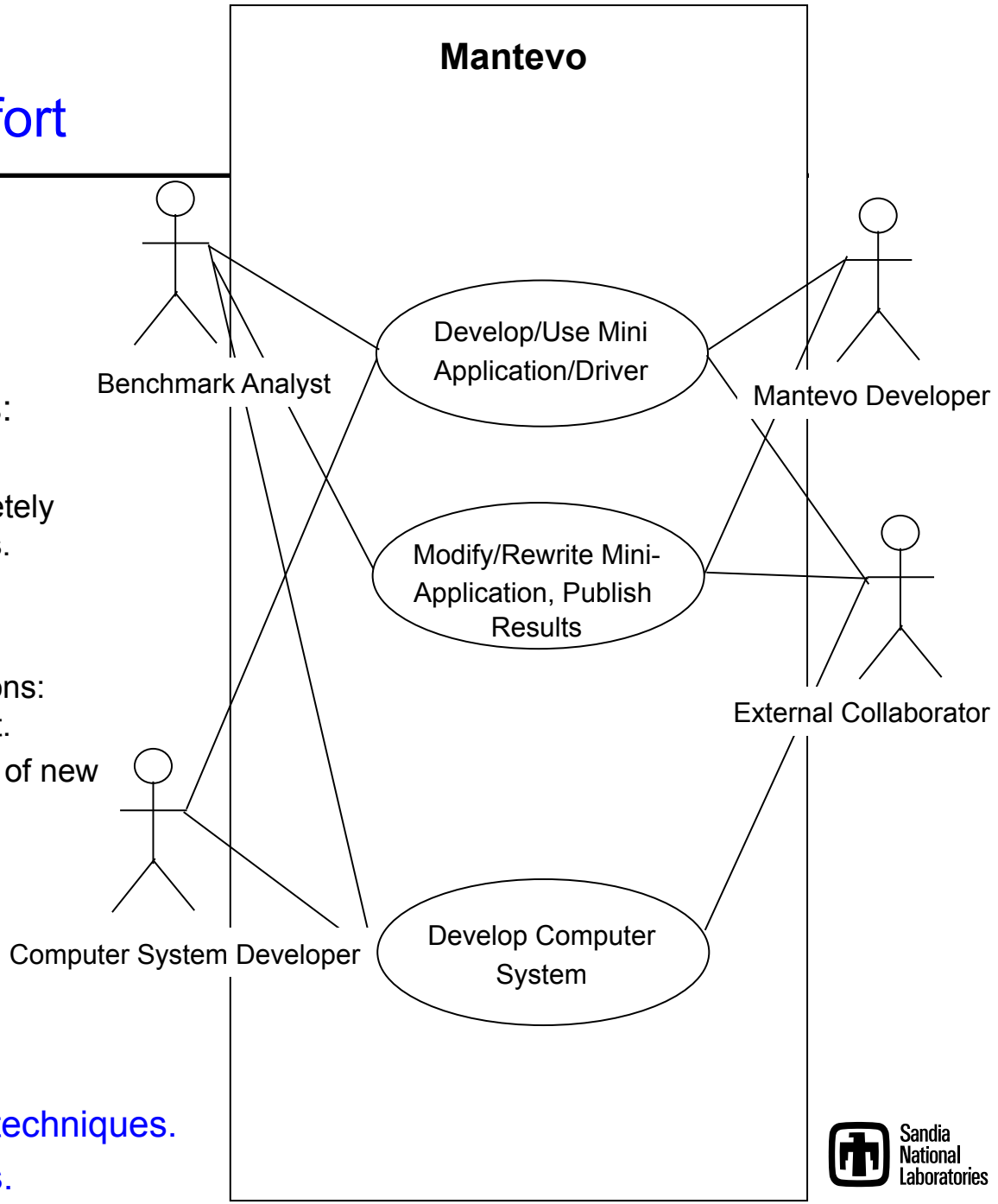
- **Goal: Develop scalable computing capabilities via:**
  - Application analysis.
  - Application improvement.
  - Computer system design.
- **Fixed timeline.**
- **Countless design decisions.**
- **Collaborative effort.**
- **Pre-Mantevo:**
  - Work with each, large application.
  - Application developers have conflicting demands:
    - Features,
    - performance.
  - Application performance profiles have similarities.



Sandia National Laboratories

Mantevo Effort

- **Develop:**
  - Mini apps, mini drivers.
- **Goals:**
  - Aid in system design decisions:
    - Proxies for real apps.
    - Easy to use, modify or completely rewrite, e.g., multicore studies.
  - Guide application and library developers:
    - Get first results in new situations: apps/libs know what to expect.
    - Better algorithms: Exploration of new approaches.
  - Predict performance of real applications in new situations.
  - New collaborations.



Results:

- Better-informed design decision.
- Broad dissemination of optimization techniques.
- Incorporation of external R&D results.

Sandia National Laboratories

Mantevo* Project

* Greek: augur, guess, predict, presage

- Multi-faceted application performance project.
- Three types of packages:
  - **Miniapps**: Small, self-contained programs.
    - **MiniFE/HPCCG**: unstructured implicit FEM/FVM.
    - **phdMesh**: explicit FEM, contact detection.
    - **MiniMD**: MD Force computations.
    - **MiniXyce**: Circuit RC ladder.
  - **Minidrivers**: Wrappers around Trilinos packages.
    - **Beam**: Intrepid+FEI+Trilinos solvers.
    - **Epetra Benchmark Tests**: Core Epetra kernels.
  - **Motif framework**: Collection of “dwarves”.
    - **Prolego**: Parameterized, composable fragment collection to mimic real apps.
- Open Source (LGPL): Fosters external collaboration.
- Staffing: Application & Library developers.

Sandia National Laboratories

Mantevo Characterization

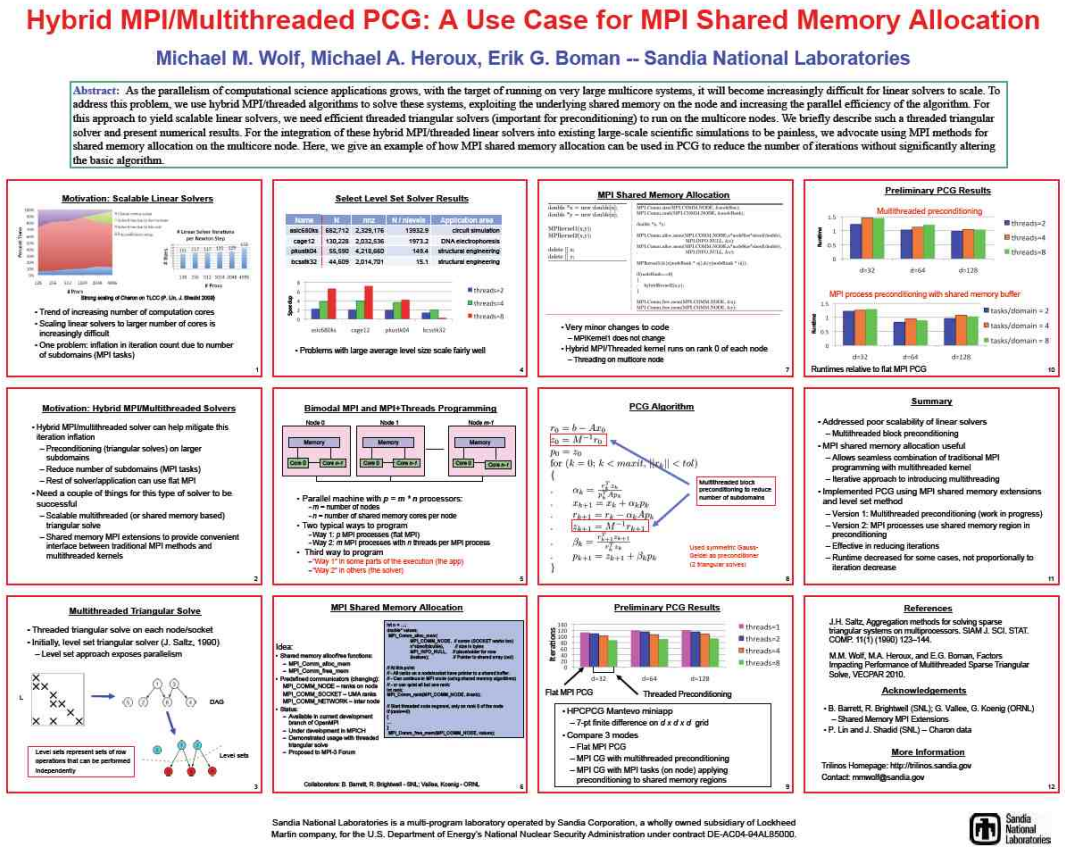
- Development of “co-design vehicles”, i.e. miniapps.
- Roles:
  - App developer: Developer & owner of miniapp (key).
  - Algorithms expert: Knowledge of algorithm options.
  - Runtime/OS expert: Knowledge of system SW.
  - HW expert: Component selection, arch trends.
  - Benchmark expert: Focused performance studies.
- Goal:
  - Concrete foundation for design studies.
    - Dwarves: “Even as cartoon characters they are sketchy.” J. Lewis.
  - Starting point for:
    - Performance studies (many kinds).
    - Algorithm replacement studies.
    - New programming models (even total rewrites).
  - Elevated conversation between all interested parties.

Sandia National Laboratories

Example Studies

Sandia National Laboratories

Bi-Modal MPI-only, MPI+threads
Michael Wolf poster



Hybrid MPI/Multithreaded PCG: A Use Case for MPI Shared Memory Allocation

Michael M. Wolf, Michael A. Heroux, Erik G. Boman -- Sandia National Laboratories

Abstract: As the parallelism of computational science applications grows, with the target of running on very large multicore systems, it will become increasingly difficult for linear solvers to scale. To address this problem, we use hybrid MPI/threaded algorithms to solve these systems, exploiting the underlying shared memory on the node and increasing the parallel efficiency of the algorithm. For this approach to yield scalable linear solvers, we need efficient threaded triangular solvers (important for preconditioning) to run on the multicore nodes. We briefly describe such a threaded triangular solver and present numerical results. For the integration of these hybrid MPI/threaded linear solvers into existing large-scale scientific simulations to be painless, we advocate using MPI methods for shared memory allocation on the multicore node. Here, we give an example of how MPI shared memory allocation can be used in PCG to reduce the number of iterations without significantly altering the basic algorithm.

Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

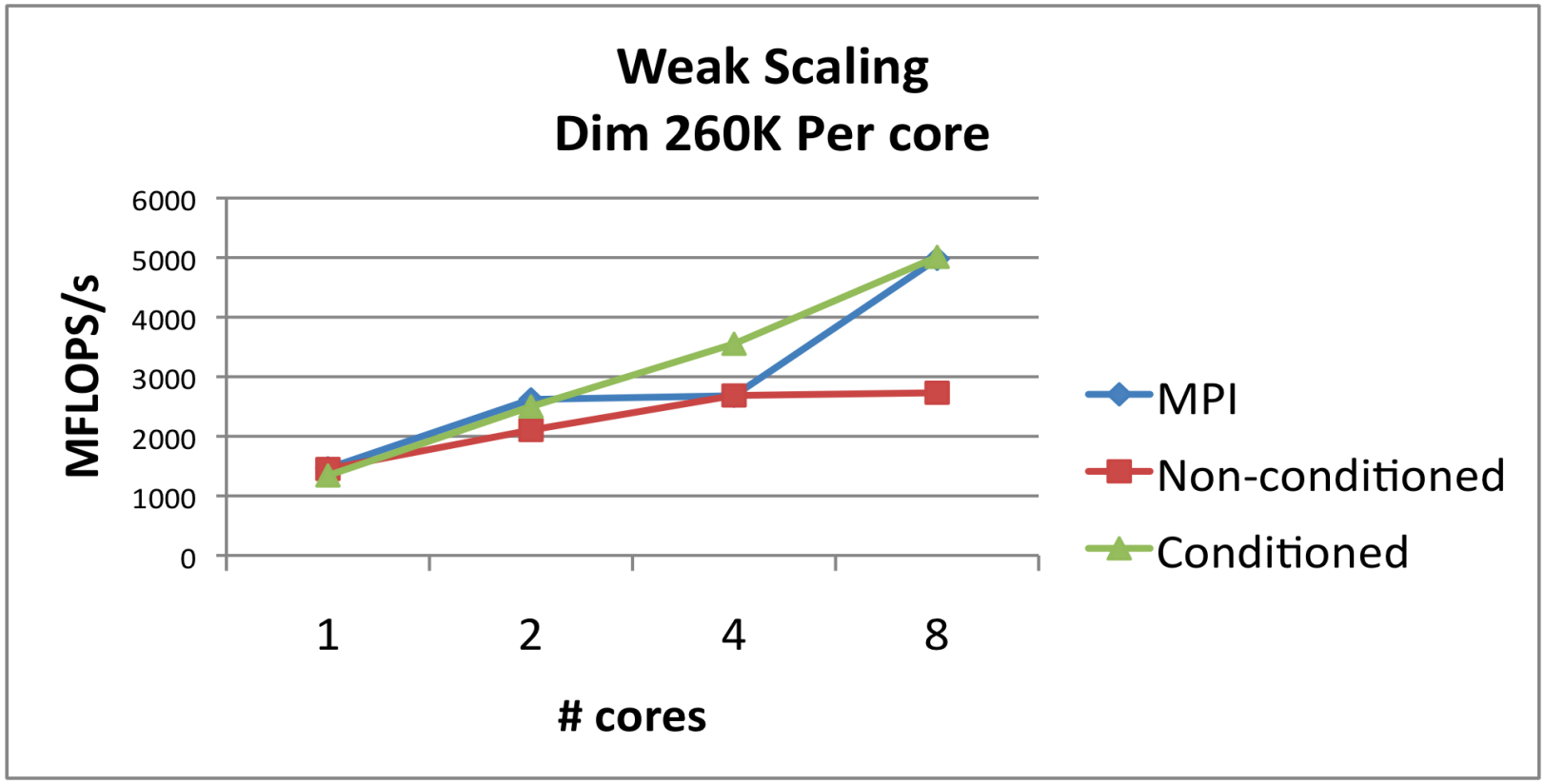
Data Placement on NUMA

- Memory Intensive computations: Page placement has huge impact.
- Most systems: First touch.
- Application data objects:
  - Phase 1: Construction phase, e.g., finite element assembly.
  - Phase 2: Use phase, e.g., linear solve.
- Problem: First touch difficult to control in phase 1.
- Idea: Page migration.
  - Not new: SGI Origin. Many old papers on topic.

Data placement experiments

- MiniApp: HPCCG
- Construct sparse linear system, solve with CG.
- Two modes:
  - Data placed by assembly, not migrated for NUMA
  - Data migrated using parallel access pattern of CG.
- Results on dual socket quad-core Nehalem system.
- Migrate-on-next-touch:
  - RT/OS feature.
  - Study: Pedretti, Merritt, *Managing Shared Memory Data Distribution in Hybrid HPC Applications,* SAND2010-6262, Sep 2010.

Weak Scaling Problem



- MPI and conditioned data approach comparable.
- Non-conditioned very poor scaling.

Much more…

- Rewrites of HPCCG:
  - Pthreads, OpenMP, Chapel, qthreads…
- MiniFE:
  - Prototype of Kokkos Node API.
  - Prototype of pipeline and task graph node parallelism.
- Performance comparisons of different platforms:
  - All.

- But: Are comparison results predictive?

Validation

Are Miniapps Predictive?

Sandia National Laboratories

Does MiniFE Predict Charon Behavior? Processor Ranking: 8 MPI tasks; 31k DOF/core

- Charon steady-state drift-diffusion BJT
- Nehalem (Intel 11.0.081 –O2 –xsse4.2; all cores of dual-socket quadcore)
- 12-core Magny-Cours (Intel 11.0.081 –O2; one socket, 4 MPI tasks/die)
- Barcelona (Intel 11.1.064 –O2; use two sockets out of the quad-socket)
- 2D Charon (3 DOF/node) vs. 3D MiniFE; match DOF/core and NNZ in matrix row
- Charon LS w/o or w/ ps: GMRES linear solve without/with ML precond setup time
- Try to compare MiniFE “assembling FE”+”imposing BC” time with Charon equivalent

MiniFE

	CG	FE assem+BC
1	Nehalem	Nehalem
2	MC(1.7)	MC(1.7)
3	Barc(2.7)	Barc(1.8)

Charon

	LS w/o ps	LS w/ ps	Mat+RHS
1	Nehalem	Nehalem	Nehalem
2	MC(1.7)	MC(1.8)	MC(1.46)
3	Barc(2.8)	Barc(2.5)	Barc(1.52)

Number in parenthesis is factor greater than #1 time

Sandia National Laboratories

MiniFE Predict Charon? Multicore Efficiency Dual-Socket 12-core Magny-Cours : 124k DOF/core

- Charon steady-state drift-diffusion BJT; Intel 11.0.081 –O2
- Weak scaling study with 124k DOF/core
- 2D Charon (3 DOF/node) vs. 3D MiniFE; match DOF/core and NNZ in matrix row
- Efficiency: ratio of 4-core time to n-core time (expressed as percentage)
- Charon LS w/o or w/ ps: GMRES linear solve without/with ML precond setup time
- 100 Krylov iterations for both MiniFE and Charon (100 per Newton step)

MiniFE

cores	CG eff
4	Ref
8	89
12	73
16	61
20	54
24	45

Charon

cores	LS w/o ps eff	LS w/ ps eff
4	Ref	Ref
8	87	89
12	74	78
16	61	66
20	49	54
24	40	45

Sandia National Laboratories

Summary

- Mantevo miniapps:
  - In many ways similar to other efforts.
  - Some strengths:
    - Completely open process: LGPL, validation.
    - Highly collaborative.
- Challenges:
  - Engaging already-busy apps developers.
  - Keeping miniapps relevant over time.
- Mantevo site: http://software.sandia.gov/mantevo

Sandia National Laboratories