Performance Engineering Research Institute (PERI)

Presented by

Patrick H. Worley

Computational Earth Sciences Group
Computer Science and Mathematics Division

Performance engineering: Enabling petascale science

Petascale computing is about delivering performance to scientists

Maximizing performance is getting harder:

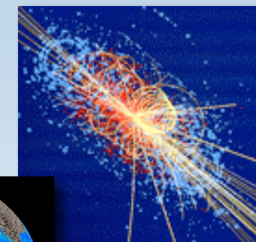
- Systems are more complicated
  - O (100 K) processors
  - Multi-core with SIMD extensions
- Scientific software is more complicated:
  - Multi-disciplinary and multi-scale

IBM BlueGene at LLNL

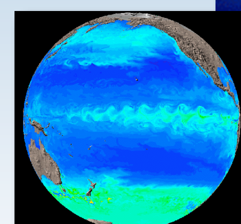
Cray XT4 at ORNL

PERI addresses this challenge in three ways:

- Model and predict application performance
- Assist SciDAC scientific code projects with performance analysis and tuning
- Investigate novel strategies for automatic performance tuning



BeamBeam3D accelerator modeling



POP model of El Nino

SciDAC Scientific Discovery through Advanced Computing

Oak Ridge National Laboratory

SciDAC-1
Performance Evaluation Research Center (PERC): 2001–2006

Initial goals:

- Develop performance-related tools and methodologies for
  - Benchmarking
  - Analysis
  - Modeling
  - Optimization

Second phase:

- In the last two years, added emphasis on optimizing performance of SciDAC applications, including
  - Community Climate System Model
  - Plasma Microturbulence Project (GYRO, GS2)
  - Omega3P accelerator model

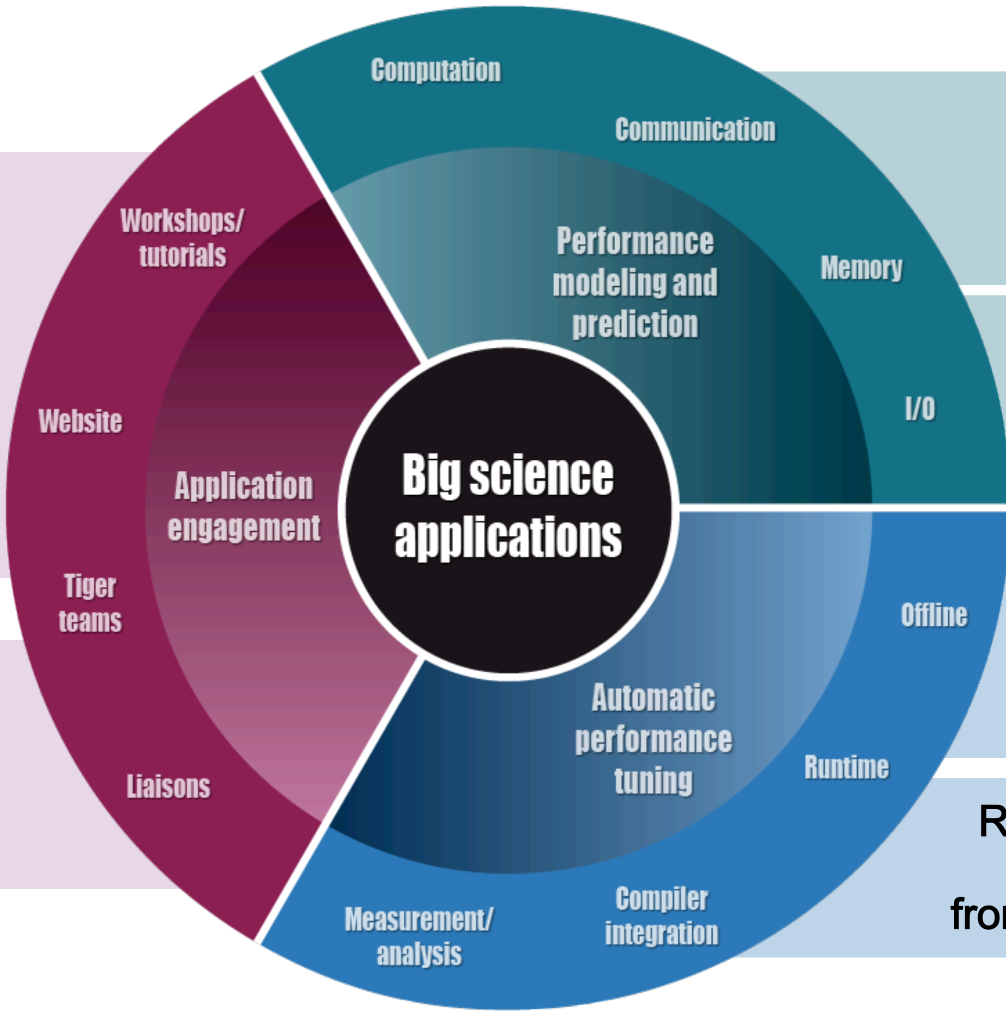
Some lessons learned

- Performance portability is critical:
  - Codes outlive computing systems.
  - Scientists can't publish that they ported and optimized code.
- Most computational scientists
  are not interested in performance tools:
  - They want performance experts to work with them.
  - Such experts are not "scalable," i.e., they are a limited resource and introduce yet another bottleneck in optimizing code.

SciDAC-2
Performance Engineering Research Institute (PERI)

Providing near-term impact on the performance optimization of SciDAC applications

Informing long-term automated tuning efforts



Evaluating architectures and algorithms in preparation for move to petascale

Providing guidance in automatic tuning

Long-term research goal to improve performance portability

Relieving the performance optimization burden from scientific programmers

Engaging SciDAC software developers

Application engagement

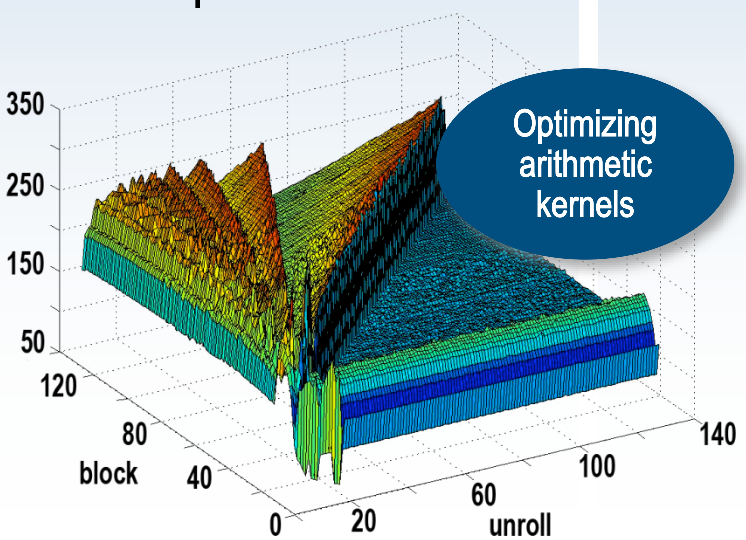
- Work directly with DOE computational scientists
- Ensure successful performance porting of scientific software
- Focus PERI research on real problems

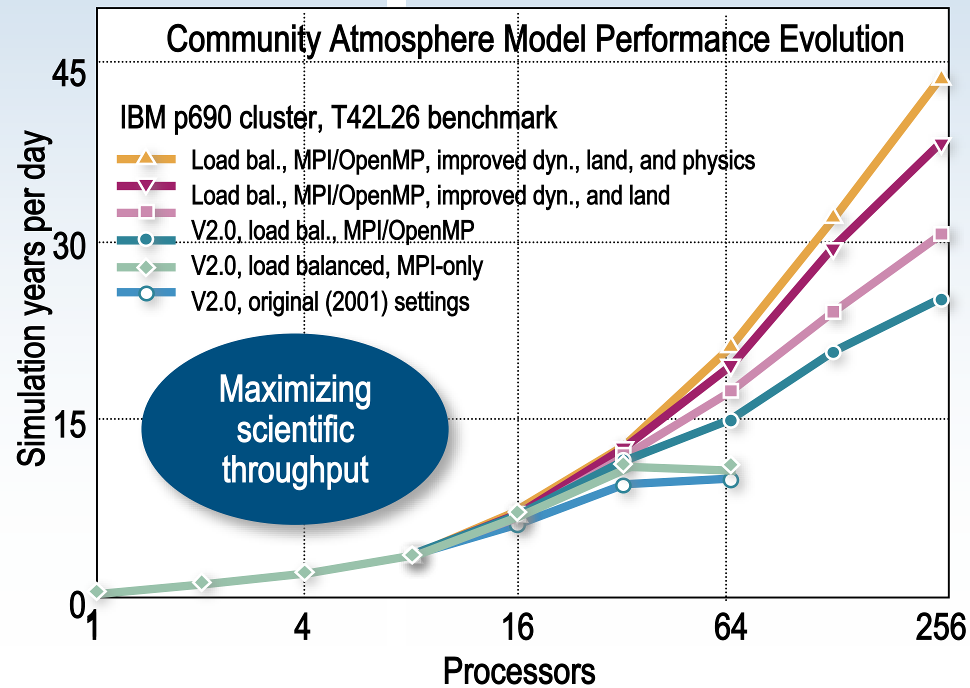
Application liaisons

- Build long-term personal relationships between PERI researchers and scientific code teams

Tiger teams

- Focus on DOE's highest priorities
  - SciDAC-2
  - INCITE
  - JOULE





SciDAC Scientific Discovery through Advanced Computing

Oak Ridge National Laboratory

FY 2007 application engagement activities

Application survey

- Collect and maintain data on SciDAC-2 and DOE INCITE code characteristics and performance requirements
- Use data to determine efficient allocation of PERI engagement resources and provide direction for PERI research
- Provide DOE with data on SciDAC-2 code portfolio

http://icl.cs.utk.edu/peri/

Application liaisons

- Active engagement (identifying and addressing significant performance issues) with five SciDAC-2 and one INCITE projects, drawn from accelerator, fusion, materials, groundwater, and nanoscience
- Passive engagement (tracking performance needs and providing advice as requested) with an additional eight SciDAC-2 projects

Tiger teams

- Working with S3D (combustion) and GTC (fusion) code teams to achieve 2007 JOULE report computer performance goals
- Tiger Team members drawn from across PERI collaboration, currently involving six of the ten PERI institutions

Performance modeling

Modeling is critical for automation of tuning:

- Guidance to the developer:
  - New algorithms, systems, etc.
- Need to know where to focus effort:
  - Where are the bottlenecks?
- Need to know when we are done:
  - How fast should we expect to go?
- Predictions for new systems.

Recent improvements:

- Reduced human/system cost.
  - Genetic Algorithms now “learn” application response to system parameters.
  - Application tracing sped up and storage requirements reduced by three orders of magnitude.
- Greater accuracy
  - S3D (combustion), AVUS (CFD), Hycom (ocean), and Overflow (CFD) codes modeled within 10% average error.

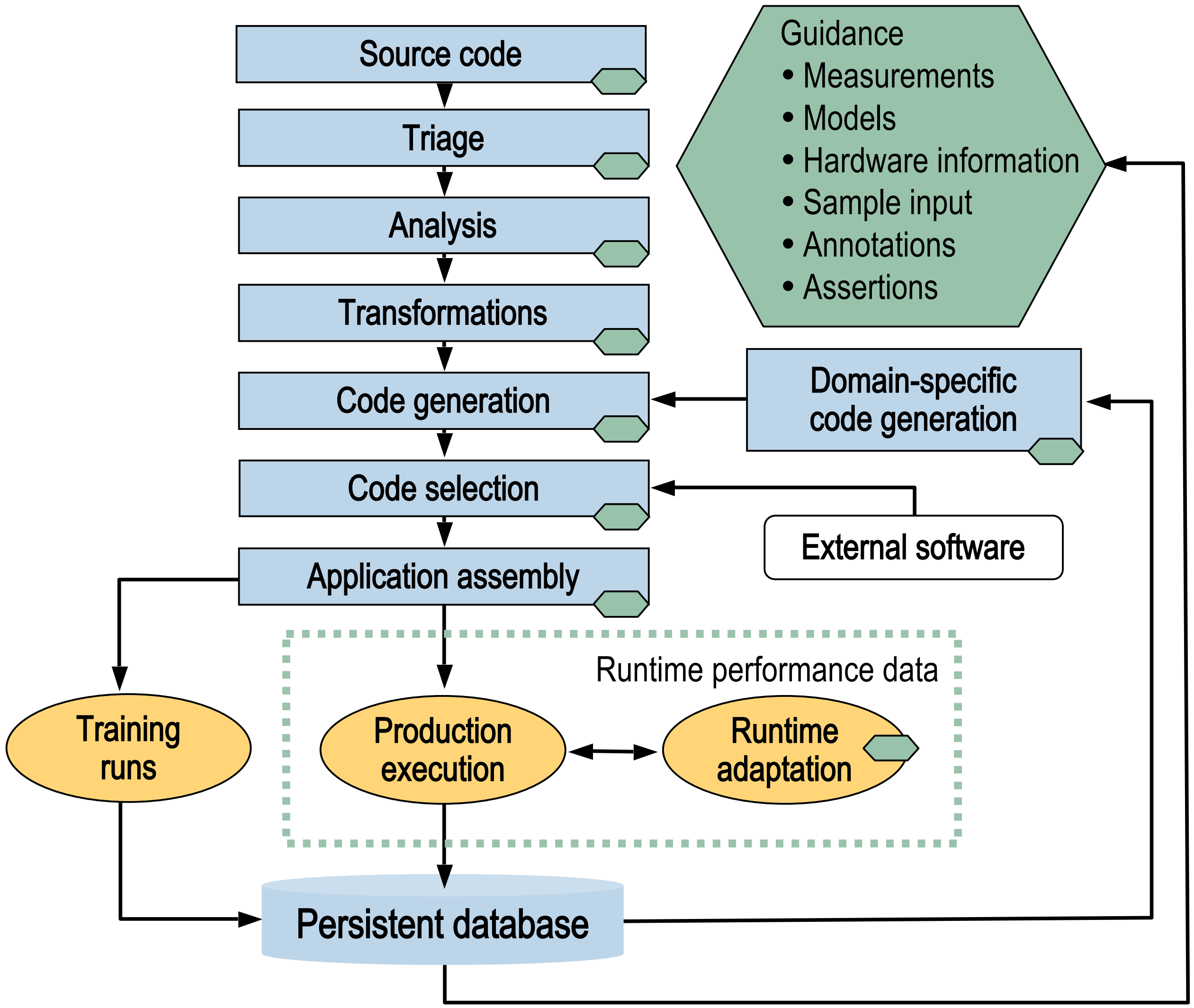
Modeling efforts contribute to procurements and other activities beyond PERI automatic tuning.

Automatic performance tuning of scientific code

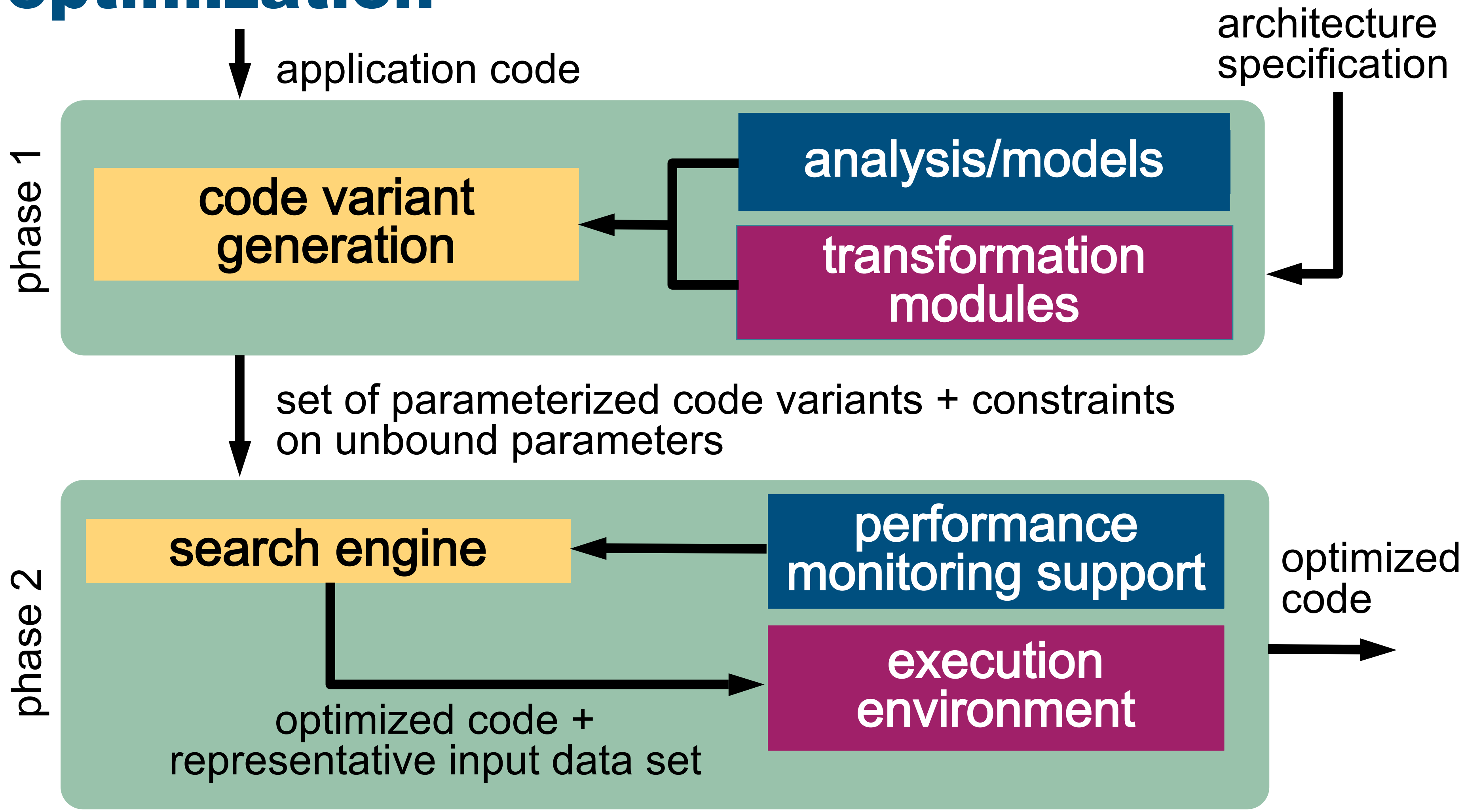
- Long-term goals for PERI
  - Obtain hand-tuned performance from automatically generated code for scientific applications
    - General loop nests
    - Key application kernels
  - Reduce the performance portability challenge facing computational scientists
    - Adapt quickly to new architectures
  - Integrate compiler-based and empirical search tools into a framework accessible to application developers
  - Runtime adaptation of performance-critical parameters

Automatic tuning flowchart

1: Triage	Where to focus effort
2: Semantic analysis	Traditional compiler analysis
3: Transformation	Code restructuring
4: Code generation	Domain-specific code
5: Code selection	Modeling and empirical search
6: Assembly	Choose the best components
7: Training runs	Performance data for feedback
8: Runtime adaptation	Optimize long-running jobs



Model-guided empirical optimization



The team

Argonne National Laboratory
Paul Hovland
Dinesh Kaushik
Boyana Norris

Lawrence Berkeley National Laboratory
David Bailey
Daniel Gunter
Katherine Yelick

Lawrence Livermore National Laboratory
Bronis de Supinski
Daniel Quinlan

Oak Ridge National Laboratory
Sadaf Alam
G. Mahinthakumar
Philip Roth
Jeffrey Vetter
Patrick Worley

Rice University
John Mellor-Crummey

University of California - San Diego
Allan Snavely
Laura Carrington

University of Maryland
Jeffrey Hollingsworth

University of North Carolina
Rob Fowler
Daniel Reed
Ying Zhang

University of Southern California
Jacqueline Chame
Mary Hall
Robert Lucas (P.I.)

University of Tennessee
Jack Dongarra
Shirley Moore
Daniel Terpstra

Contacts

Patrick H. Worley

Computational Earth Sciences Group
Computer Science and Mathematics Division
(865) 574-3128
worleyph@ornl.gov

Fred Johnson

DOE Program Manager
Office of Advanced Scientific Computing Research
DOE Office of Science

Oak Ridge National Laboratory