Mini-Symposium on Scientific Software Engineering
Application software development at extreme-scale: challenges and opportunities

“While computer performance has improved dramatically, real productivity in terms of the science accomplished with these ever-faster machines has not kept pace. Indeed, scientists are finding it increasingly costly and time consuming to write, port, or rewrite their software to take advantage of the new hardware.”

Stuart Faulk – productivity gridlock

Where are the productivity bottlenecks?
Productive Computing Goal

- Maximizing the value, reducing the cost, and minimizing the effort needed to achieve the solution to a complex problem through the application of modern supercomputing.

**Time to discovery**

- Physics –1
- Physics –2
- Physics –n

**Software Development**

- Code Optimization
- Model Validation

**Compilation**

**Execution**

- Auto-tuning

**Scientific Insights**

Software development time

Software execution time
The time and effort needed to specify, develop, test, tune, execute, and maintain large community-based application software greatly impact the overall productivity of computational science.
IDEAS Project Background

**Workshop Finding**
- Scientific software engineering R&D
- Software performance productivity metrics
- Software as a scientific virtual facility
- Easy to use software tools to support reusability, portability, and inter-operability
- Strategies for scientific sustainability
- Tools for legacy code understanding and transformation
- Easy to use software development tools
- Community code development/management

**IDEAS Project**
- Co-development effort jointly funded by ASCR and BER
- Multidisciplinary team (applied math, computer architecture, domain science)
- Use-case to test and validate new software engineering approaches
- Outreach to DOE computational science communities
Biological and Environmental Research

Understanding complex biological, climatic, and environmental systems across vast spatial and temporal scales
Institutional Leads (Pictured)
Full Team List

Science Use Cases
J. David Moulton
Tim Scheibe
Carl Steefel
Glenn Hammond
Reed Maxwell
Scott Painter
Ethan Coon
Xiaofan Yang

Hans Johansen
Lois Curfman McInnes
Ross Bartlett
Todd Gamblin*
Andy Salinger*
Jason Sarich
Jim Willenbring
Pat McCormick

Methodologies for Software Productivity

Use Cases: Terrestrial Modeling
Software Productivity for Extreme-Scale Science
Methodologies for Software Productivity
Extreme-Scale Scientific Software Development Kit (xSDK)

Project Leads
ASCRI: M. Heroux and L.C. McInnes
BER: J. D. Moulton

Extreme-scale Scientific Software Development Kit (xSDK)
Mike Heroux
Ulrike Meier Yang
Jed Brown
Irina Demeshko
Kirsten Kleese van Dam
Sherry Li
Daniel Osei-Kuffuor
Vijay Mahadevan
Barry Smith

Outreach
David Bernholdt
Katie Antypas*
Lisa Childers*
Judith Hill*