Overview of the IDEAS Software Productivity Project

Mike Heroux (SNL), Lois Curfman McInnes (ANL), J. David Moulton (LANL), David Bernholdt (ORNL), Hans Johansen (LBNL)
And all IDEAS project members
**Confluence of trends**

- **Fundamental trends:**
  - Disruptive HW changes: Requires thorough algorithm/code refactoring
  - Demands for coupling: Multiphysics, multiscale

- **Challenges:**
  - Need refactorings: Really, continuous change
  - Modest app development funding: No monolithic apps
  - Requirements are unfolding, evolving, not fully known a priori

- **Opportunities:**
  - Better design and SW practices & tools are available
  - Better SW architectures: Toolkits, libraries, frameworks

- **Basic strategy:** Focus on **productivity**
Interoperable Design of Extreme-scale Application Software (IDEAS)

Motivation
Enable increased scientific productivity, realizing the potential of extreme-scale computing, through a new interdisciplinary and agile approach to the scientific software ecosystem.

Objectives
Address confluence of trends in hardware and increasing demands for predictive multiscale, multiphysics simulations.
Respond to trend of continuous refactoring with efficient agile software engineering methodologies and improved software design.

Impact on Applications & Programs
Terrestrial ecosystem use cases tie IDEAS to modeling and simulation goals in two Science Focus Area (SFA) programs and both Next Generation Ecosystem Experiment (NGEE) programs in DOE Biologic and Environmental Research (BER).

Approach
ASCER/BER partnership ensures delivery of both crosscutting methodologies and metrics with impact on real application and programs.

Interdisciplinary multi-lab team (ANL, LANL, LBNL, LLNL, ORNL, PNNL, SNL)

ASCER Co-Leads: Mike Heroux (SNL) and Lois Curfman McInnes (ANL)
BER Lead: David Moulton (LANL)
Topic Leads: David Bernholdt (ORNL) and Hans Johansen (LBNL)
Integration and synergistic advances in three communities deliver scientific productivity; outreach establishes a new holistic perspective for the broader scientific community.

www.ideas-productivity.org
IDEAS project structure and interactions

**DOE Program Managers**
- **ASCR:** Thomas Ndousse-Fetter
- **BER:** Paul Bayer, David Lesmes

**IDEAS: Interoperable Design of Extreme-scale Application Software**
- **ASCR Co-Leads:** Mike Heroux (SNL) and Lois Curfman McInnes (ANL)
- **BER Lead:** J. David Moulton (LANL)

**Executive Advisory Board**
- John Cary (Tech-X)
- Mike Glass (SNL)
- Susan Hubbard (LBNL)
- Doug Kothe (ORNL)
- Sandy Landsberg (DOD)
- Paul Messina (ANL)

**DOE Program Managers**
- **ASCR:**
  - Thomas Ndousse-Fetter
- **BER:**
  - Paul Bayer, David Lesmes

**Executive Advisory Board**
- John Cary (Tech-X)
- Mike Glass (SNL)
- Susan Hubbard (LBNL)
- Doug Kothe (ORNL)
- Sandy Landsberg (DOD)
- Paul Messina (ANL)

**BER Use Cases**
- Lead: J. David Moulton (LANL)
- Carl Steefel (LBNL) *1
- Scott Painter (ORNL) *2
- Reed Maxwell (CSM) *3
- Glenn Hammond (SNL)
- Tim Scheibe (PNNL)
- Laura Condon (CSM)
- Ethan Coon (LANL)
- Dipankar Dwivedi (LBNL)
- Jeff Johnson (LANL)
- Eugene Kikinzon (LANL)
- Sergi Molins (LBNL)
- Steve Smith (LLNL)
- Carol Woodward (LLNL)
- Xiaofan Yang (PNNL)

**Methodologies for Software Productivity**
- Lead: Mike Heroux (SNL)
- Roscoe Bartlett (ORNL)
- Todd Gamblin* (LLNL)
- Christos Kartsaklis (ORNL)
- Pat McCormick (LANL)
- Sri Hari Krishna Narayanan (ANL)
- Andrew Salinger* (SNL)
- Jason Sarich (ANL)
- Dali Wang (ORNL)
- Jim Willenbring (SNL)

**Extreme-Scale Scientific Software Development Kit**
- Lead: Lois Curfman McInnes (ANL)
- Jed Brown* (ANL)
- Irina Demeshko (SNL)
- Anshu Dubey (LBNL)
- Alicia Klinvex (SNL)
- Sherry Li (LBNL)
- Vijay Mahadevan (ANL)
- Daniel Osei-Kuffuor (LLNL)
- Barry Smith (ANL)
- Matthew Thomas (PNNL)
- Ulrike Yang (LLNL)

**Outreach and Community**
- Lead: David Bernholdt (ORNL)
- Katie Antypas* (NERSC)
- Lisa Childers* (ALCF)
- Judy Hill* (OLCF)
- Bill Spotz* (SNL)

**Crosscutting Lead:** Hans Johansen (LBNL)

**SFAs**
**CLM**
**ACME**
**NGEE**
**Exascale Co-Design**
**Exascale Roadmap**
**ASCR Math & CS**
**ScIDAC**
**NERSC**
**OLCF**

**BER Terrestrial Programs**
**DOE Extreme-scale Programs**
**DOE Computing Facilities**
Use cases: Multiscale, multiphysics representation of watershed dynamics

- **Use Case 1**: Hydrological and biogeochemical cycling in the Colorado River System
- **Use Case 2**: Thermal hydrology and carbon cycling in tundra at the Barrow Environmental Observatory
- **Use Case 3**: Hydrologic, land surface, and atmospheric process coupling over the continental United States

- Leverage and complement existing SBR and TES programs:
  - LBNL and PNNL SFAs
  - NGEE Arctic and Tropics

- **General approach**:
  - Leverage existing open source application codes
  - Improve software development practices
  - Targeted refactoring of interfaces, data structures, key components to facilitate interoperability
  - Modernize management of multiphysics integration and multiscale coupling
IDEAS interconnections

- **Use cases:** Drive efforts. Traceability from all efforts
  - But generalized for future efforts

- **Methodologies ("HowTo") for SWP:**
  - Infrastructure, testing, porting, refactoring, portability, etc.
  - Workflows, lifecycles: Document and formalize. Identify best practices

- **xSDK:** frameworks + components + libraries
  - Build apps by aggregation and composition

- **Outreach:** Foster communication, adoption, interaction

- **First of a kind:** Focus on **software productivity**
Software engineering for extreme-scale science

- Methodology-related issues for extreme-scale software:
  - Ineffective code and project infrastructure?
  - Supporting science/library testing?
  - Heroic, risky porting/refactoring?
  - Performance portability?

- “Pre-” application readiness
  - Software Engineering issues that slow down science
  - These issues are beyond training on performance tools and new hardware
Goal: Put steps in place to encourage adoption and reuse of research libraries, and improve longevity of ASCR software investments through refactoring and interoperability.
IDEAS role: Methodologies, gaps, glue

Use Cases: Terrestrial Modeling

App 1
App 2
App 3
App ...

Team 1
Team 2
Team ...

xSDK 1
xSDK 2
xSDK ...

HW 1
SW 1
Tool 1
Tool 2
SW 2
HW 2

Extreme-scale Platforms & Facilities

PRODUCTIVITY “GAP”
- Readiness
- Porting
- Reuse
- Staffing

PRODUCTIVITY “OPPORTUNITY”
- Readiness – from teams/community
- Porting – testing, refactoring tools
- Reuse – xSDK, interoperability
- Staffing – “How To” resources, knowledge from extended teams
Motivation: Scientific software teams have a wide range of levels of maturity in software engineering practices

- Baseline survey of xSDK and BER Use Case teams

Approach:

- ‘What Is’ docs: 2-page characterizations of important software project topics
- ‘How To’ docs: brief sketch of best practices
  - Emphasis on “bite-sized” topics enables CSE software teams to consider improvements at a small but impactful scale.

Initial emphasis:

- What is CSE Software Productivity?
- What are Software Testing Practices?
- How to Add and Improve Testing in Your CSE Software Project

Topics in progress:

- Refactoring tools and approaches
- Best practices for using interoperable libraries
- Designing for performance portability
- Etc.

Impact: Provide baseline nomenclature and foundation for next steps in SW productivity and SW engineering for CSE teams
Managing issues: Fundamental software process

- Issue: Bug report, feature request

- Approaches:
  - Short-term memory, office notepad
  - ToDo.txt on computer desktop (1 person)
  - Issues.txt in repository root (small co-located team)
  - ...
  - Web-based tool + Kanban (distributed, larger team)
  - Web-based tool + Scrum (full-time dev team)

- IDEAS project:
  - Jira Agile + Confluence: Turnkey web platform (ACME too)
  - Kanban: Simplest of widely known Agile SW dev processes
Kanban principles

- Limit number of “In Progress” tasks
- Productivity improvement:
  - Optimize “flexibility vs swap overhead” balance. No overcommitting.
  - Productivity weakness exposed as bottleneck. Team must identify and fix the bottleneck.
  - Effective in R&D setting. Avoids a deadline-based approach. Deadlines are dealt with in a different way.
- Provides a board for viewing and managing issues
Managing IDEAS Project Activities Using JIRA Agile and Kanban

Kanban Board, on Jira site.
Four columns:
- To Do
- Selected
- In Progress
- Done
**xSDK focus**

- **Common configure and link capabilities**
  - xSDK users need full and consistent access to all xSDK capabilities
  - Namespace and version conflicts make simultaneous build/link of xSDK difficult
  - Determining an approach that can be adopted by any library or components development team for standardized configure/link processes

- **Library interoperability**

- **Designing for performance portability**

```plaintext
domain components
- reacting flow, etc.
- reusable.

libraries
- solvers, etc.
- interoperable.

frameworks & tools
- doc generators.
- test, build framework.

sw engineering
- productivity tools.
- models, processes.
```

Extreme-scale scientific software development kit (xSDK)
**Standard xSDK package installation interface**

**Motivation:** Obtaining, configuring, and installing multiple independent software packages is tedious and error prone.
- Need consistency of compiler (+version, options), 3rd-party packages, etc.

**Approach:** Define a standard xSDK package installation interface to which all xSDK packages will subscribe and be tested

**Accomplishments:**
- Work on implementations of the standard by the hypre, PETSc, SuperLU, and Trilinos developers
- PETSc can now use the “scriptable” feature of the installers to simultaneously install hypre, PETSc, SuperLU, Trilinos with consistent compilers and ‘helper’ libraries.

**xSDK Build Example**

**Impact:** Foundational step toward seamless combined use of xSDK libraries, as needed by BER use cases and other multiphysics apps.
Enabling Interoperable Biogeochemistry with Alquimia

Several geochemistry libraries are well established in the community making geochemistry ideal to explore componentization and interface design. **Alquimia is an interface library, and does not perform any reaction calculations.**

- Alquimia currently assumes reactive transport uses operator-splitting.
- Fully-implicit reactive transport support is being developed in collaboration with IDEAS.
- Assists in enforcing geochemical conditions (speciation) for transport boundary conditions.
- Alquimia can facilitate benchmarking of geochemical capabilities in existing codes.
- Geochemistry libraries, such as PFLOTRAN and CrunchFlow, have implemented interfaces to Alquimia.

Alquimia is open source, [https://bitbucket.org/berkeleylab/alquimia](https://bitbucket.org/berkeleylab/alquimia)
Increased BER investment: Use Case 3 now starting:
Hydrologic, land surface, & atmospheric process coupling over the contiguous United States

Feedbacks between ground- and surface-water dynamics, vegetation processes, and the atmospheric boundary layer significantly affect local and regional climate across a range of scales and climatic conditions.

- Type and distribution of vegetation cover affects
  - Land-atmosphere water and energy fluxes
  - Runoff generation and overland flow processes
- Variability in evapotranspiration is strongly dependent on lateral ground-water flow.
- Impact of mountainous terrain is not well understood.

Explicit high-resolution modeling of groundwater-land surface-atmosphere feedbacks is necessary to understand and predict hydrologic response.
Use Case 3: Synergies with IDEAS

- Exploring **componentization and interoperability**
  - Analysis and possible refactoring of existing data structures in ParFlow
  - Componentization of integrated hydrology
  - Release land model components as a modular library

- Establishing new **community benchmarks** for relevant large-scale integrated hydrology simulations of terrestrial systems.

- Establishing a strong connection to **performance portability** objectives, with new simulations running on more cores, for longer times, and producing more data.
Outreach and community

- **Objective:** Begin changing the way computational and domain science communities think about software development through increased awareness, education, and active engagement

- Building connections to computing facilities

- Presentations and posters at BER PI meetings, SE-HPCCSE14 workshop, organized SIAM CSE 2015 mini-symposium

- Leadership roles in NITRD CSESSP (sustainability and productivity) workshop (Oct 2015)

- BOF proposal for SC15 with international colleagues

**Software productivity**

discussion at OLCF Annual User Meeting, June 2015

CSESSP workshop homepage, October 2015
Better software productivity is essential for extreme-scale CSE

- **Better SW productivity can give us better, faster and cheaper**
  - **Better**: Science, portability, robustness, composability
  - **Faster**: Execution, development, dissemination
  - **Cheaper**: Fewer staff hours and lines of code

- **IDEAS project**
  - Enabling production of high-quality science results, rapidly and efficiently
    - Multiscale terrestrial ecosystem science
    - Broadly: DOE extreme-scale scientific apps
  - Delivering first-of-a-kind extreme-scale scientific software ecosystem
    - xSDK
    - SWP methodologies (“HowTo”)
    - Outreach and community

Essential mechanism for progress
- In time of disruptive change
- In presence of multiple design tradeoffs
Expanding on IDEAS

- **IDEAS currently comprises two elements**
  - Core R&D on software engineering and productivity issues (comp. sci./appl. math)
    - Producing direct results and general resources (WhatIs, HowTos, other documents, tutorials, etc.)
  - Science application “use cases” motivate and validate CS/AM productivity R&D
    - Assistance with adoption too (implicit in validation)

- **Possible scaling approaches in funded programs**
  - Expand CS/AM work on SWE/SWP (currently only scratching the surface)
  - Add other applications/domains (new use cases or independent projects)
    - May motivate different CS/AM focus areas
  - Program-level SWE/SWP resources, including assistance with adoption

- **Building and engaging a broader community**
  - These issues are not unique to DOE or to US
  - Establishing CSE Software Forum ([cse-software.org](http://cse-software.org)) to organize larger community
    - Facilitate broader contribution to and sharing of SWE/SWP activities and resources
    - Facilitate leveraging work of others, but not a substitute for DOE investment
    - Will need to show compelling nucleus of people and resources to attract serious participation
    - Simple starting point: join the cse-forum mailing list on http://cse-software.org
Computational Science & Engineering (CSE) Software Forum

Who are we?

The Computational Science & Engineering (CSE) Software Forum is an international group of researchers and other stakeholders from national laboratories, academic institutions, and industry who are interested in (and concerned about) development processes for software for computational science and engineering and related technical computing areas, with a particular interest in CSE on high-performance (parallel) computers.

What are we doing?

We're creating a clearinghouse to gather, discuss, and disseminate experiences, techniques, tools, and other resources for the entire lifecycle of CSE software, from initial concept through the growth phase to maturity, re-investment, and eventually to retirement.

We want to raise awareness of the importance of good software practices to scientific productivity and to the quality and reliability of computationally-based scientific results, as well as raising awareness of the increasing challenges facing CSE software developers as high-end computing heads to extreme scales.
Engaging with IDEAS

- Talk to us: ideas-leads@lists.anl.mcs.gov
- Visit our website: http://ideas-productivity.org
- Join the community: go to http://cse-software.org and join the cse-forum mailing list (or cse-announce)