We are Failing to Meet National Needs

• Although the numbers have been rising recently, the number of undergraduate and graduate degrees in awarded in science and engineering are not meeting national needs.

Source: KEY SCIENCE AND ENGINEERING INDICATORS: 2010 DIGEST
Filling the Gap

• We are increasingly dependent on foreign born graduates to fill the gap especially at the graduate level
Problems Attracting and Retaining Students

• Students switching to non-science majors
  – Over 90% indicated poor instruction among reasons for switching
  – 26% had trouble learning the basic concepts (Seymour and Hewitt, 1997)

• Engineering program at Wright State University improved the four-year graduation rate by 40% by introducing an applied mathematics course to precede the traditional calculus sequence

• Recent National Research Council report on undergraduate physics education recognized the importance of inquiry-based methods in retaining students by improving student’s conceptual understanding
The Need For Computational Scientists

• A number of national studies document the need for computational scientists
  
  – “…computer modeling and simulation are the key elements for achieving progress in engineering and science.” NSF Blue Ribbon Panel on Simulation-Based Engineering Science
  
  – “A persistent pattern of subcritical funding overall for SBE&S threatens U.S. leadership and continued needed advances…” International Assessment Of Research And Development In Simulation-Based Engineering And Science
  
  – Nearly 100% of the respondents indicated that HPC tools are indispensable, stating that they would not exist as a viable business without them or that they simply could not compete effectively. IDC Study for Council on Competitiveness of Chief Technology Officers of 33 Major Industrial Firms
Computational Science Requires New Skills

• In science and engineering, modeling and simulation is used to explore complex systems, design and test new products, and provide new insights.

• In social sciences, computational methods are used to simulate individual and group behaviors, forecast economic and demographic trends, and spatial interactions.

• In the arts, digital technologies have revolutionized design, animation, and artistic creations.

• In the humanities, digital technologies are being used to explore historical landscapes, create, explore, and analyze collections and archives.
Examples of Modeling Problems

• Tracing the spread and evolution of disease (http://supramap.osu.edu/)
• Collaborations to explore historical and contemporary events and social interaction (http://www.ichass.illinois.edu/Projects/Projects.html)
• Predicting the impacts of earthquakes (http://nees.org/)
• Designing and testing new nanomaterials and devices (http://nanohub.org/)
• Discovering oil reserves (http://access.ncsa.illinois.edu/Stories/oil/)
• Designing new packaging (http://phx.corporate-ir.net/phoenix.zhtml?c=104574&p=irol-newsArticle&ID=651774&highlight)
• Discovering how the brain works (http://www.compete.org/publications/detail/503/breakthroughs-in-brain-research-with-high-performance-computing/)
XSEDE Education Program Goals

• Prepare the current and next generation of researchers, educators and practitioners.
• Create a significantly larger and more diverse workforce in STEM.
• Inculcate the use of digital services as part of their routine practice for advancing scientific discovery.
XSEDE Education Program Services

- Campus Visits
- Assistance with program creation
- Workshops for faculty and students
- Repository of shared materials
- Other resources
Initiating Services to Facilitate Change

- Campus visits
  - First discussions about integrating computational science into the curriculum
  - Discussion of formal programs
  - Opportunities for faculty professional development
  - Overview of related XSEDE services
Promoting Formal Academic Programs

- XSEDE Education program is focused on assisting with the initiation and enhancement of formal computational science and engineering programs
  - Both undergraduate and graduate programs
  - Most sustainable way to help achieve the long-term project goals by producing a savvy workforce
  - Reduce the barriers to program adoption by
    - Providing program models
    - Solidifying a virtual community to share experiences
    - Providing faculty professional development
Creating Computational Science Programs

• Inherently interdisciplinary
  – Science, engineering, social science, or humanities domain
  – Mathematics
  – Computer science

• Expertise often dispersed across multiple departments, colleges, institutions

• Difficulty of negotiating requirements, responsibilities, and institutional arrangements
Model Computational Science Programs

• Model programs can be used to shorten the time it takes for institutions to implement new programs

• Models based on experience in creating programs in Ohio and elsewhere
  – Developed competencies needed for programs focused on science and engineering
  – Now adding competencies on data analysis, archiving, and management required across science and humanities

• Used to guide implementation of curriculum changes

• Bridges disciplinary boundaries
Undergraduate minor program overview

- Undergraduate minor program
  - 4-6 courses
  - For majors in variety of fields
- Faculty defined competencies for all students
- Reviewed by business advisory committee
- Currently being updated to reflect changes in hardware and software technologies
- Requirements adjusted to reflect the needs of different majors

<table>
<thead>
<tr>
<th>Competencies for Undergraduate Minor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation and Modeling</td>
</tr>
<tr>
<td>Programming and Algorithms</td>
</tr>
<tr>
<td>Differential Equations and Discrete Dynamical Systems</td>
</tr>
<tr>
<td>Numerical Methods</td>
</tr>
<tr>
<td>Optimization</td>
</tr>
<tr>
<td>Parallel Programming</td>
</tr>
<tr>
<td>Scientific Visualization</td>
</tr>
<tr>
<td>One discipline specific course</td>
</tr>
<tr>
<td>Capstone Research/Internship Experience</td>
</tr>
</tbody>
</table>
Flexibility in Implementation

• Adapt existing courses by adding computationally oriented modules
• Discipline oriented courses dependent on existing faculty expertise and interests
• Different subsets of required and optional competencies tied to major, required math, and example projects
Graduate Level Competencies

• Assumes some of the background of an undergraduate
• Focus more on research skills
• Core areas focus on the computer science and related modeling skills
• Need to branch into a wider array of specializations based on the nature of the graduate program
Developing Faculty Expertise

• Faculty professional development workshops
  – Two to six day workshops on a variety of topics
    • Computational thinking
    • Computational science education in science and engineering domains
  – Focus on local/regional audiences to reduce travel costs
  – Subsidies for faculty to travel to workshops at other sites
Special Workshops for Faculty and Students

• Development of synchronous and asynchronous education and training sessions
  – Multi-site broadcasts of workshops
  – Online training and education modules
  – Experimenting with full courses that can be widely shared for credit and non-credit inclusion in curricula (e.g. https://www.xsede.org/xsede-offers-free-online-parallel-computing-course)
Repository of Shared Materials

• Developing a repository of computational science education materials
  – Reviewed by professional staff and faculty
  – Indexed by subject and a detailed competency-based ontology
  – Goal: trusted, comprehensive source of information for computational science educators
  – http://hpcuniversity.org/resources/search/
Some Other Opportunities

• Journal of Computational Science Education
  – www.jocse.org
  – Peer reviewed article on computational science education experiences

• Become a reviewer or contributor to the online repository

• Use the XSEDE online materials
  – www.xsede.org
Questions and Discussion