Struggles at the Frontiers of Workforce Development:

... Acquisition and Development of Software Intensive Systems

Software Engineering Institute
Carnegie Mellon University

Dr. Kenneth E. Nidiffer
SEI Background

Funded by the U.S. government as a research & development lab (FFRDC)
Created in 1984 and administered by Carnegie Mellon University
Headquartered in Pittsburgh, Pennsylvania; offices and support worldwide
Content

- Struggles Impacting Workforce Development at the Software Frontiers
- Challenges of Workforce Development for IT and Systems Engineering
- Seventeen Workforce Issues for Developing Software-Intensive Systems

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers

Essential Difficulties

According to Fred Brooks,* software projects are difficult because of accidental and essential difficulties

• Accidental difficulties are caused by the current state of our understanding
  – of methods, tools, and techniques
  – of the underlying technology base

• Essential difficulties are caused by the inherent nature of software
  – invisibility – lack of physical properties
  – Complexity – for its size
  – conformity
  – changeability

* The Mythical Man-Month by Fred Brooks, Addison Wesley, 1995
Struggles Impacting Workforce Development at the Software Frontiers

Essential Difficulties

Complexity:

• Due to interaction of components, number of possible states grows much faster than lines of code
• For its size, software is very complex compared to other engineering artifacts
• Hardware is complex, but the laws of physical science usually tell us what to expect for a known input

Source SEI
Essential Difficulties - Complexity

- The flowchart might correspond to a 100 LOC module with a single loop that may be executed no more than 20 times.
- There are approximately $10^{14}$ possible paths that may be executed!
- For any but the smallest programs, complete path coverage for defect detection is impractical.
- Limited natural governance

Lehman Laws:

1. The Law of Continuing Change – programs must change to be useful
2. The Law of Increasing Complexity – programs that change become more complex

Struggles Impacting Workforce Development at the Software Frontiers - **Infancy of Software Engineering**

Source: SEI
## Struggles Impacting Workforce Development at the Software Frontiers - Infancy of Software Engineering

<table>
<thead>
<tr>
<th></th>
<th>Physical Science</th>
<th>Bioscience</th>
<th>Computer/Software/Cyber Science</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Origins/History</strong></td>
<td>Begun in antiquity</td>
<td>Begun in antiquity</td>
<td>Mid-20th Century</td>
</tr>
<tr>
<td><strong>Enduring Laws</strong></td>
<td>Laws are foundational to furthering exploration in the science</td>
<td>Laws are foundational to furthering exploration in the science</td>
<td>Only mathematical laws have proven foundational to computation</td>
</tr>
</tbody>
</table>
| **Framework of Scientific Study** | Four main areas: astronomy, physics, chemistry, and earth sciences | Science of dealing with health maintenance and disease prevention/treatment | • Several areas of study: computer science, software/systems engineering, IT, HCI, social dynamics, AI  
• All nodes attached to/relying on netted system |
| **R&D and Launch Cycle** | 10-20 years | 10-20 years | Significantly compressed; solution time to market needs to happen very quickly |

Source: SEI

HCI: Human Computer Interaction; AI: Artificial intelligence
Struggles Impacting Workforce Development at the Software Frontiers - Demographics of Workforce Provide Different Views of the Frontiers

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - Demographics of Workforce Provide Different Views of the Frontiers

- Demographics of workforce are changing, and different views may emerge with multiple generations to consider.
- Generation Y professionals are technically savvy and can better leverage IT capabilities for improved efficiencies and productivity; however, may lack the systems engineering knowledge, skills, and abilities.

Sources: SEI, Recommendations for Improving Acquisition Training, May 2010
Achieving Effective Acquisition of Information Technology in the Department of Defense, National Academy of Sciences, 2010
Struggles Impacting Workforce Development at the Software Frontiers - **Software Is Everywhere with Limited Natural Governance**

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - **Software Is Everywhere with Limited Natural Governance**

Laws of physics
Laws of software
Challenge of algorithms
Difficulty of distribution & concurrency
Problems of design
Importance of organization
Impact of economics
Influence of politics
**Limits of human imagination**

Source: IBM
Struggles Impacting Workforce Development at the Software Frontiers - **Software Is Increasingly Complex**

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - Software Is Increasingly Complex

Increasing Functionality and Complexity:

- System of systems
- Architecture
- Services
- Networked platforms
- People who digitally connect to cyberspace

Transportation Infrastructure + Healthcare Infrastructure + Banking & Financial Infrastructure + Energy & Utilities Infrastructure + Communications Infrastructure

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - Software Is Increasingly Complex

Increasing Use of Innovative Processes, Methods and Tools

**Predictive Models**
- Requirements
- Design
- Implementation
- Verification
- Maintenance

**Iterative Models**
- Waterfall
- Spiral
- RAD
- RUP

**Adaptive Models**
- Scrum
- XP
- OpenUP

Source: Noblis

Dr. Ken Nidiffer
October 2014
© 2014 Carnegie Mellon University
Struggles Impacting Workforce Development at the Software Frontiers - **Software Connects Us in Near Real Time**

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - **Software Is Becoming a More Personal and Valued Utility**

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - **Software Is Globally Important**

Manufacturing

Finance

Space and Aviation

Engineering

Source: SEI
Struggles Impacting Workforce Development at the Software Frontiers - Software Is Today’s Strategic Resource
Challenges of Workforce Development for IT and Systems Engineering
Challenges of Workforce Development for IT and Systems Engineering

Building out capabilities to manage large information technology projects has been a sore spot for the Air Force.*

Specifically, the service has been challenged with developing IT acquisition talent among its ranks, adopting and maintaining processes that foster best practices and aligning acquisition and cybersecurity strategies.*

*Lt. Gen. Charles Davis, the military deputy in the Office of the Assistant Secretary of the Air Force for Acquisition

Source: C4ISR & Networks, Feb 2014
Challenges of Workforce Development for IT and Systems Engineering

Software Proves Great Capabilities, but Struggling, Bifurcated Communities

Source: SEI
Challenges of Workforce Development for IT and Systems Engineering

Defense Acquisition Reform: Where Do We Go from Here? Permanent Subcommittee on Investigations, United States Senate; October 2, 2014
Seventeen Workforce Issues for Developing Software-Intensive Systems
Need to Meet the Workforce Challenges

Large U.S. Government Programs Often Stumble

Source: SEI
Seventeen Current Struggles in Developing Software Intensive Systems

1. Software requirements often change during a software project as knowledge is gained and the scope of the project and the product emerge.

2. Requirements for new and modified software often influence, and are influenced by, an organization’s business processes and employees’ workflow processes of employees.

3. Intellectual capital of software personnel is the primary capital asset for software projects and software development organizations because software is a direct product of human cognitive processes.

4. Communication and coordination within software teams and with project stakeholders often lack clarity.

* Software Extension to the PMBOK
Seventeen Current Struggles in Developing Software Intensive Systems

5. Creation of software requires innovative problem solving to create unique solutions. Software projects are more akin to research and development projects than to construction or manufacturing projects.

6. Exhaustive testing of software is impractical because of the time that would be required to test all logical paths and interfaces under all combinations of input data and stimuli.

7. Software development often involves inclusion of different vendor products and development of interfaces to other software; this may result in integration and performance issues.

* Software Extension to the PMBOK
Seventeen Current Struggles in Developing Software Intensive Systems

8. Software projects involve risk and uncertainty because they require innovation, the product is intangible, and stakeholders may not effectively articulate, or agree on, the needs to be satisfied by the software product.

9. Planning and estimation for software projects is challenging because they depend on requirements, which are often imprecise, and on historical data, which is often missing or inapplicable. Preparing accurate estimates is also challenging because the efficiency and effectiveness of software developers are widely variable.

10. Product complexity makes development and modification of software challenging because of the enormous number of logical paths within program modules, the number of data values that exercise the paths, and the combinations of interface details among program modules.
Seventeen Current Struggles in Developing Software Intensive Systems

11. Because most software is interconnected, information security techniques are necessary. Software security is a large and growing challenge.

12. Objective quantification and measurement of software quality are difficult because of the intangible nature of software.

13. Software developers use processes, methods, and tools that are constantly evolving and frequently updated.

14. Software is often the element of a system that is changed when functionality, behavior, or quality attributes must be changed.

15. A software product may be required to operate on a variety of hardware platforms and infrastructure software.

* Software Extension to the PMBOK
Seventeen Current Struggles in Developing Software Intensive Systems

16. Executable software is not a standalone product. It is executed on computing hardware and is often an element of a system consisting of diverse hardware, other software, and manual procedures.

17. Platform technologies, infrastructure software, and vendor-supplied software are frequently changed or updated, which can necessitate changes to the software being developed.

* Software Extension to the PMBOK
Questions?
Contact Information

Dr. Kenneth E. Nidiffer, Director of Strategic Plans for Government Programs

Software Engineering Institute, Carnegie Mellon University
Office: +1 703-247-1387
Fax: +1 703-908-9235
Email: Nidiffer@sei.cmu.edu