Preparing the Ground for Next Generation Software Engineering

Don O’Neill
Email: ONeillDon@aol.com

State of the Industry
The May 2005 Report of the 2nd National Software Summit (NSS2) entitled “Software 2015: A National Software Strategy to Ensure U.S. Security and Competitiveness” [1] lays out a ten-year concept plan with the vision: “Achieving the ability to routinely develop trustworthy software products and systems, while ensuring the continued competitiveness of the U.S. software industry.” The plan includes 11 significant initiatives within four major program areas:

1. Improving Software Trustworthiness
2. Educating and Fielding the Software Workforce
3. Re-Energizing Software Research and Development
4. Encouraging Innovation Within the U.S. Software Industry

New Issues and Challenges
New issues are now emerging surrounding the production, fielding, and operation of net-centric systems of systems that are [2], [3]:

1. Essential to the competitiveness and security of the nation's critical infrastructure,
2. Essential to the defense and security of the Global Information Grid, and
3. Essential to the offense and security of Cyber Power strategies.

Unclaimed Benefits and Unmet Needs
However, there remain unclaimed benefits and unmet needs stemming from earlier neglect [4]. The immediate goal of practical Next Generation Software Engineering is to drive systems and software engineering to do more with less... fast. Four practical objectives are identified to advance the goal using smart and trusted technologies:

1. Drive user domain awareness
2. Simplify and produce systems and software using a shortened development life cycle
3. Compose and field trustworthy applications and systems from parts
4. Compose and operate resilient systems of systems from systems

More specifically:
1. Driving user domain awareness calls for integrating systems, software, and user needs and synthesizing mission needs in terms of systems, software and user; improving user domain awareness maturity and conducting user domain awareness assessments; and exploiting NGSE technology through interactive virtual user experience and simulation.

2. Simplifying and producing systems and software using a shortened development life cycle calls for eliminating bottlenecks through automation of labor-intensive activities; accelerating delivery through Wiki-based requirements, incremental development, and Agile approaches; exploiting NGSE technology through formality in requirements expression and smart compilers; and measuring speed and trustworthiness [5].

3. Composing and fielding trustworthy applications and systems from parts calls for managing rapid release through aspect-based commitment management, fact-based
aspect and attribute assurance, and real-time risk management; focusing on supplier assurance through process maturity, global supply chain management, and configuration management; exploiting NGSE technology through attribute-based architecture, smart middleware, interoperability, intrusion detection, intrusion protection, and intrusion tolerance; and measuring frequency of release and trustworthiness [5].

4. Composing and operating resilient systems of systems from systems calls for exercising control before, during, and after adversity; focusing on situation awareness through intelligent middlemen and information sharing; exploiting NGSE technology through coordinated recovery time objectives, distributed supervisory control, and operation sensing and monitoring; and measuring control and resilience [6], [7].

In managing the investment needed to meet these objectives, capability portfolio investments are organized by management, process, and engineering. For best results, the implementation transition through table 1 is to begin in the Northwest corner and move towards the Southeast. In this way user domain awareness, shortened life cycle, systems from parts, and systems of system from systems provide a natural spiral of incremental activities where current work in progress builds on preceding work accomplished.

<table>
<thead>
<tr>
<th>Practical Next Generation Software Engineering Goal</th>
<th>Management</th>
<th>Process</th>
<th>Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objective 1</strong></td>
<td>Drive user domain awareness towards more harmonious cooperation among people and machines</td>
<td>Integrate needs of systems, software, and user</td>
<td>User domain awareness maturity</td>
</tr>
<tr>
<td><strong>Strategic Measures</strong></td>
<td>1. User satisfaction</td>
<td>• Synthesize mission needs in terms of systems, software, and user</td>
<td>• Assessment of user domain awareness</td>
</tr>
<tr>
<td></td>
<td>2. Trustworthiness</td>
<td>• Apply team innovation management</td>
<td></td>
</tr>
<tr>
<td><strong>Objective 2</strong></td>
<td>Simplify and produce systems and software using a shortened development life cycle</td>
<td>Eliminate bottlenecks</td>
<td>Accelerate delivery</td>
</tr>
<tr>
<td><strong>Strategic Measures</strong></td>
<td>1. Speed</td>
<td>• Automation of labor-intensive activities</td>
<td>• Wiki-based requirements</td>
</tr>
<tr>
<td></td>
<td>2. Trustworthiness</td>
<td></td>
<td>• Incremental development</td>
</tr>
<tr>
<td><strong>Objective 3</strong></td>
<td>Compose and field trustworthy applications</td>
<td>Rapid release</td>
<td>Supplier Assurance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Aspect-based</td>
<td></td>
</tr>
</tbody>
</table>
and systems from parts

<table>
<thead>
<tr>
<th>Strategic Measures</th>
<th>Control</th>
<th>Awareness</th>
<th>NGSE technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Frequency of release</td>
<td>• Exercise control</td>
<td>• Intelligent middlemen</td>
<td>• Coordinated recovery time objectives</td>
</tr>
<tr>
<td>2. Trustworthiness</td>
<td>• Fact-based aspect and attribute assurance</td>
<td>• Information sharing</td>
<td>• Distributed supervisory control</td>
</tr>
<tr>
<td></td>
<td>• Real-time risk management</td>
<td>• Situation awareness</td>
<td>• Operation sensing and monitoring</td>
</tr>
<tr>
<td></td>
<td>• Process maturity</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Global supply chain management</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Configuration management</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In progressing through the dimensions of management, process, and engineering each industry sector, application, system, and system of systems brings with it its own context and culture. The manner by which a community of interest addresses these practical Next Generation Software Engineering objectives is influenced by the domain engineering paradigms, management and engineering processes, fielding and operating practices, government regulations, and public expectation to which it responds.

**Conclusion**

Driving user domain awareness towards more harmonious cooperation among people and machines in systems acquisition is an imperative. Without this awareness throughout the life cycle and across the functional domains of acquisition management, program management, systems engineering and software engineering, the gap between user expectation and user satisfaction will continue to grow and mission execution will suffer. With this awareness user engineering, software engineering, and systems engineering will be better aligned; the synergy between user considerations and software will be better expressed in Next Generation Software Engineering approaches; and the intersectional innovation resulting from cross discipline clash will impact systems acquisition and the missions it supports.

**References**


This article is part of the IEEE Reliability Society 2008 Annual Technology Report.


https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/business/1016-BSI.html

http://iac.dtic.mil/dacs