

National Agenda for Software Engineering Research & Development: *Architecting the Systems of the Future*

July 2021

Dr. Forrest Shull

Lead for Defense Software Acquisition Policy Research,
Carnegie Mellon University / Software Engineering Institute

President,
IEEE Computer Society

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Copyright 2021 Carnegie Mellon University.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

DM21-0617

Outline

CMU/SEI National Agenda for Software Engineering Study:

- Motivation
- Methodology
- 10-15 Year Research Roadmap
- Recommendations



CMU SEI - a US Federally Funded Research and Development Center



Our mission: Engineering and securing software

Established in 1984 at Carnegie Mellon University

US FFRDC charged to improve the state of the practice of software engineering and cyber security

Focus of National Agenda for Software Engineering

Software is vital to America's **global competitiveness**, **innovation**, and **national security**. The economy, the nation's infrastructure, education, and healthcare all depend on software.



Lead a community effort to:

1. **Identify future challenges** in engineering software-reliant systems.
2. Develop a **research roadmap** that will drive advances in **foundational software engineering principles** across system types such as intelligent, autonomous, safety-critical, and data intensive systems.
3. Raise the **visibility** of software to the point where it receives sustained recognition commensurate with its importance to national security and competitiveness.
4. Enable strategic partnerships and collaborations to **drive innovation among industry, academia, and government**.

New System Types Require New R&D

New types of systems

- very adaptive defense mission systems
- systems that perform data fusion at a huge scale
- highly engineered business enterprises
- personal digital assistants—that really assist
- dynamically integrated healthcare
- smart cities, buildings, roads, cars, and transport

Trends

- scale motivating the need for safe and resilient software composition
- rapid and assured continuous software evolution
- addressing workforce gaps in software talent
- artificial intelligence (AI)-inspired automation
- evidence-based assurance that a system behaves as intended
- impact of cyber-social platforms on social behavior, creating societal scale impact
- primacy of data (data has become as important as code)



Approach



- **Advisory Board**



- **Computing Landscape**



- **Emerging Technologies**



- **Literature Review**

- **Expert Interviews**



- **Workshops**

- **Future Scenarios**

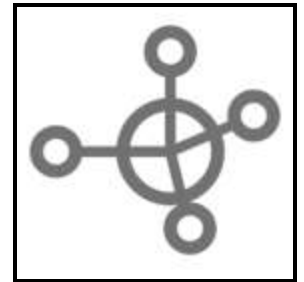
Cast a wide net with input from many communities

**National Agenda Study:
Roadmap / Outcome**



Codify findings in an actionable way

Diverse Ecosystem



Ecosystem acts on findings

A Small Subset of Example Participants

Academia / Govt Research

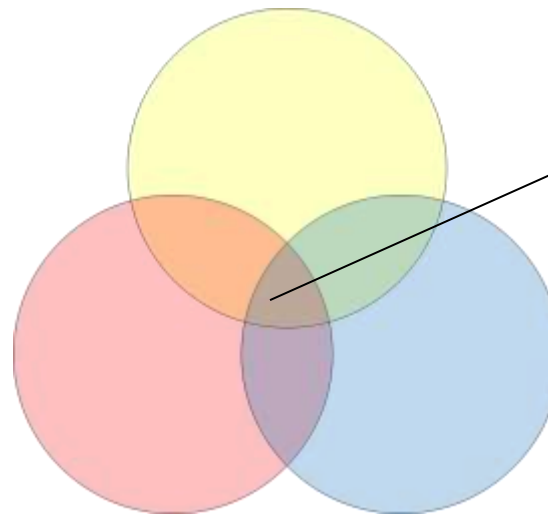
- **Dr. Sergey Bratus**, DARPA PM, Information Innovation Office; winner of 2013 BlackHat Pwnie award for Most Innovative Research
- **Dr. Hal Finkel**, Program Manager in DOE's Office of Science, Advanced Scientific Computing Research (ASCR)
- **Dr. Sol Greenspan**, NSF Program Director for Software and Hardware Foundations (SHF), Cyberinfrastructure for Sustained Scientific Innovation (CSSI), and others
- **Dr. Jim Herbsleb**, Director, Institute for Software Research, CMU
- **Dr. Laurie Williams**, Distinguished Professor, North Carolina State University (NCSU); Winner of the ACM SIGSOFT Influential Educator Award

Industry

- **Mr. Mark Boyd**, Senior Technical Fellow in Software and Chief Software Engineer in Boeing AvionX
- **Mr. Patrick Lardieri**, Lockheed Martin Fellow for Cyber
- **Dr. John Launchbury**, Chief Scientist, Galois, Inc
- **Mr. Jacob Torrey**, Manager, Secure Hardware and Foundational Technologies, Amazon Web Services (AWS)
- **Robin Yeman**, Lockheed Martin Space Senior Fellow

Mission Experts

- **Dr. Jared Dunnmon**, Technical Director, AI/ML, at Defense Innovation Unit (DIU)
- **Dr. Amy Henninger**, Senior Advisor for Software and Cybersecurity at DOT&E
- **Ms. Hannah Hunt**, Chief Product and Innovation Officer at Army Futures Command Software Factory
- **Dr. David Martinez**, Laboratory Fellow, MIT Lincoln Labs



These groups, and especially their intersection, help to identify gaps and provide important insights

Emerging Vision of the Future of Software Engineering

The current notion of software development will be replaced by one where **the software pipeline consists of humans and AI as trustworthy collaborators that rapidly evolve systems based on user intent.**

Advanced development paradigms lead to efficiency and trust at scale.

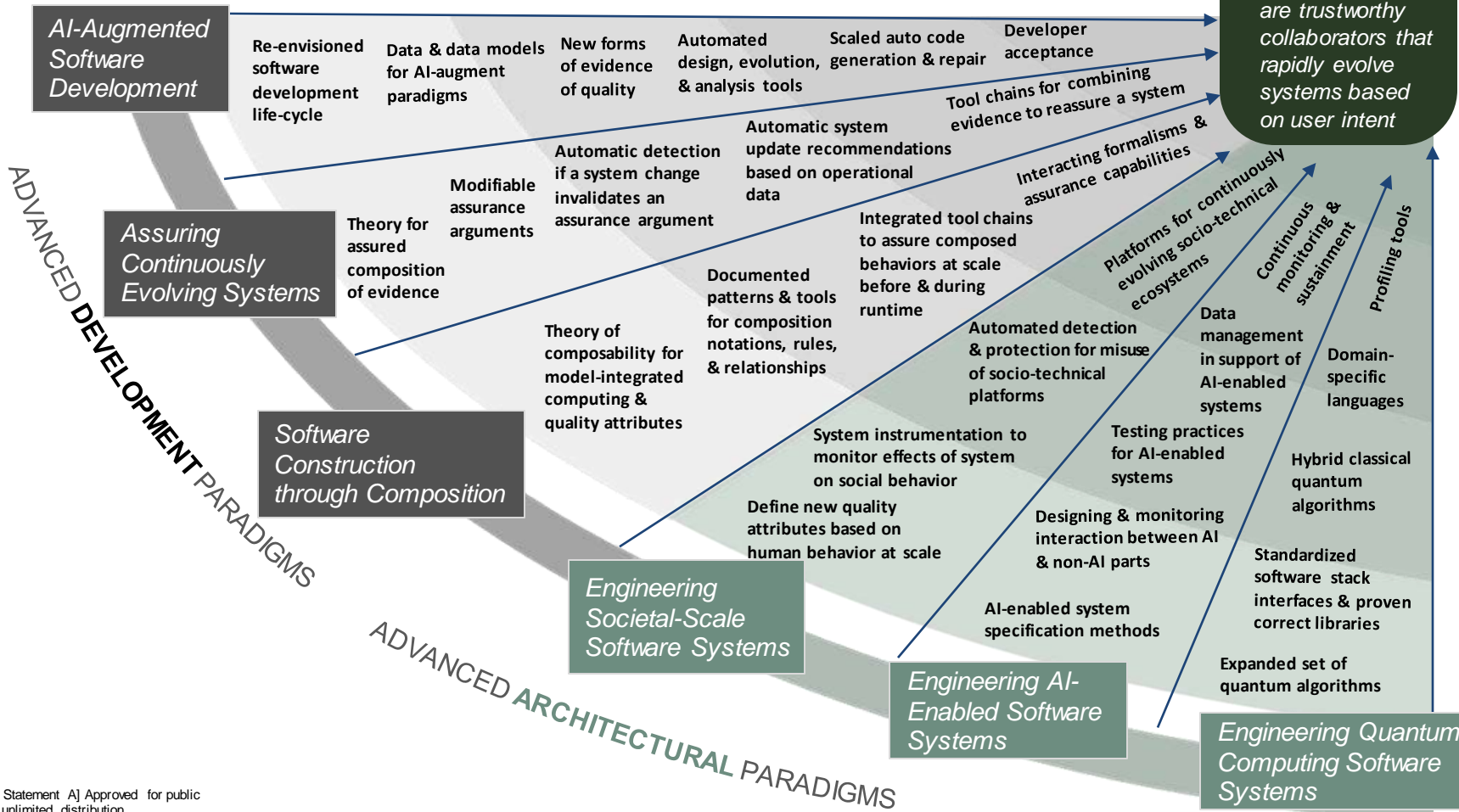
- Humans leverage trusted AI as a workforce multiplier for all aspects of software creation.
- Formal assurance arguments are evolved to assure and efficiently re-assure continuously evolving software.
- Advanced software composition mechanisms enable predictable construction of systems at increasingly large scale.



Advanced architectural paradigms enable the predictable use of new computational models.

- Theories and techniques drawn from the behavioral sciences are used to design large-scale socio-technical systems, leading to predictable social outcomes.
- New analysis and design methods facilitate the development of quantum-enabled systems.
- AI and non-AI components interact in predictable ways to achieve enhanced mission, societal, and business goals.

Software Engineering Research Roadmap with Research Focus Areas and Research Objectives (10-15 Year Horizon)



[Distribution Statement A] Approved for public release and unlimited distribution.

Research Focus Areas: Development Paradigms

AI-Augmented Software Development

The focus of this research area is on what AI-augmented software development will look like at each stage of the development process and during continuous evolution, where it will be particularly useful in taking on routine tasks.

Assuring Continuously Evolving Systems

The goal of this research area is to develop a theory and practice of rapid and assured software evolution that enables efficient and bounded reassurance of continuously evolving systems.

Software Construction through Composition

This research area focuses on methods and tools that enable the specification and enforcement of composition rules for component-based technologies and platforms that allow both the creation of required behaviors and the assurance of these behaviors.

Research Focus Areas: Architectural Paradigms

Engineering Societal-Scale Software Systems

This research area leverages the social sciences to develop new software engineering approaches that enable predictable behavior of software systems consisting of people as system components.

Engineering AI-Enabled Software Systems

This research area focuses on exploring which existing software engineering practices can reliably support the development of AI systems, as well as identifying and augmenting software engineering techniques for systems with AI components.

Engineering Quantum Computing Software Systems

Our goals in this research area are to first enable current quantum computers to be easily programmed, and then enable increasing abstraction as larger, fully fault-tolerant quantum computing systems become available.

Findings

1. Software engineering **profoundly impacts all aspects of society** as we increasingly rely on it to provide complex and critical functionality.

2. **Key issues** impacting future directions in software engineering include **smart automation, reassuring evolving systems, understanding composed systems**, and **new system types**.

3. Research focused on **integrating heterogeneous systems at ever-larger scales** is needed to support the emergence of new system types.

4. Beyond scale, tomorrow's software challenges will include the need to address **social software considerations** such as transparency, freedom from bias, and privacy.

5. **Academia, government, industry, and research labs** will have critically important and **interacting roles** for the future of software engineering.

6. Significant **research opportunities lie at the intersection** of software engineering and other fields.

7. Software engineering is a distributed activity, taking place worldwide. Technologies are needed to support **seamless integration across computing environments and teams**.

8. **New approaches** are needed to meet growing needs in the **workforce**.

Selected Recommendations

Our goal is to catalyze change that advances software engineering, which in turn will lead to more trustworthy and capable software-reliant systems in the future. Advancing software engineering requires supportive policy, research funding, researchers, practitioners, and cross-fertilization with other research communities.

Accordingly, we make the following recommendations:

- 1 Increase investment in software engineering research.**
- 2 Initiate software engineering grand challenges.**
- 3 Increase software engineering workforce quantity and effectiveness.**

1 **Recommendation:** Increase Investment in software engineering research.

- A. Long-term investment to fund core programs in software engineering research is needed to enable new partnerships and advance software engineering technologies to build the systems of the future. Software engineering research is foundational to build the software supply chain including the tools, techniques and software infrastructure to build future software-reliant systems.
 - i. Funding for software engineering research seems out of step with the criticality of software and with the research challenges to position for the future.
 - ii. Software industry partnerships with national labs and academic research communities sponsored through government investment enable effective collaboration to solve new problems and provide the long-term progress needed to meet the future needs of software engineering
 - iii. Research sponsors prioritize funding of software engineering research, including the technical focus areas identified in this study. The technical focus areas are foundational areas that amplify industry research activities, enabling broad advancement in software engineering.

2 **Recommendation:** Initiate software engineering grand challenges.

- A. Government sponsorship of 2-3 software engineering grand challenges will further technological innovation and inspire new partnerships to advance software engineering.

Grand challenges have become an effective way to quickly mobilize existing capability on critical issues while enabling new partnerships across academia and industry.

3 **Recommendation:** Increase software engineering workforce quantity and effectiveness.

- A. New approaches are needed for talent and workforce development to significantly expand the scale of software engineering talent required to meet future needs. Multiple actions are necessary to continue to build software engineering capacity, including:
- i. continuous education, including some training capability built into systems through digital assistants
 - ii. research and training on how to maximize automation in software engineering activities to best utilize the time of people
 - iii. incentives for individuals and organizations to support training and education at all levels, from high school to undergraduate and graduate education
 - iv. emphasis on creating effective tools for non-software engineering professionals to create automated solutions that are safe, reliable, secure, and resilient

Summary

- New types of systems will continue to push beyond the bounds of what current software engineering theories, tools, and practices can support, therefore **CMU SEI has been leading a study that aims to capture emerging shifts in how software is developed.**
- **Future systems** and **fundamental shifts** in software engineering **require new research focus.**
 - Rapidly deploying software with confidence will be critical for addressing needs and expectations
 - Greater **speed in translating technology into fielded capability** is vital for achieving and maintaining **technological edge.**
- Tomorrow's software challenges will include the need to address new "-ilities": social software considerations such as transparency, freedom from bias, and privacy.
- New approaches are needed for **talent and workforce development** to significantly expand the scale of software engineering talent required to meet future needs.



Discussion

