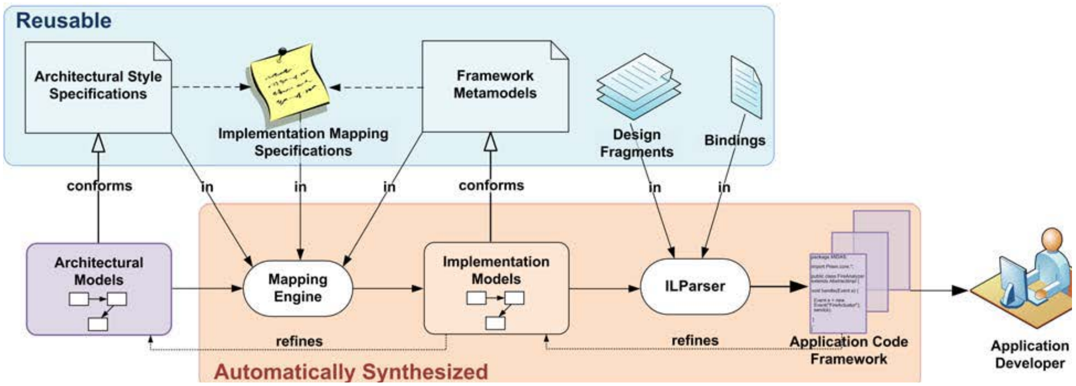
# Software Engineering Group




My research and educational, and interests are in the design and engineering of complex software-intensive systems. I work mainly in the research fields known as *software engineering* and *systems engineering*. Today's ultra-large-scale cyber-social systems truly demand novel design and engineering approaches: approaches that integrate computational, physical, and human/social concerns. I am especially interested in how technical decisions influence business, social, and other forms of economic value. I seek innovations to reduce the cost, increase the quality, and extend the reach of software. My projects with my graduate students span from the economics of design (options value of modularity), to automated software development (model-driven software engineering), to the formal modeling of design spaces and tradeoffs analysis, to ultra-large-scale systems approaches to creating a future, national-scale learning healthcare system for the United States.


## Kevin Sullivan

Associate Professor
sullivan@cs.virginia.edu
www.cs.virginia.edu/people/faculty/sullivan.html

Department of Computer Science
85 Engineer's Way
Charlottesville, VA 22904-4740
sullivan@virginia.edu


"Integrate and evolve."

University of Virginia

School of Engineering & Applied Science

## Software Engineering

Software engineering is the research discipline that seeks to produce value for businesses, people, and society by developing and evaluating novel methods, processes, and tools to reduce the cost, improve the quality, and extend the applications of software to important systems of all kinds. The field combines roots in computer science (e.g., logic and proof systems, programming language semantics, static and dynamic program analysis) with a systems engineering perspective that embraces the full life-cycle and context of software and software-driven systems. Systems-level issues include requirements, economics, the physical aspects of software-controlled systems, and the nature, roles, and goals of people and institutions within and around such systems. My group is open to addressing research problems in this broad space.

## Ultra-Large-Scale Systems

Ordinary systems are complex at scales that can be managed by traditional engineering methods of rigorous, centralized control of system design and development. Computers, software, and networks, however, have made it possible to connect arbitrarily complex, distributed, and heterogeneous systems into what are now being called *ultra-large-scale (ULS) systems*. If traditional systems are like buildings, ULS systems are like cities. One can create infrastructure, services, governance mechanisms, etc., but what then happens is under no one entity's control. Rather, a ULS system comprises innumerable autonomous, often self-interested, parties working in ways that, if properly conceived, constrained, and supported, produce wonderful emergent results. We are now in an era of ULS systems. I was deeply involved with, and co-authored, the seminal report published by the Carnegie Mellon Software Engineering Institute entitled, *Ultra-Large-Scale Systems: The Software Challenge of the Future* (2006), which established that traditional (software) engineering methods and mindsets are inadequate to address the nature and needs of such systems. In the last years, the Institute of Medicine and the Office of the National Coordinator of the U.S. Department of Health and Human Services have embraced this perspective as a key to realizing one of their most critical goals, a national-scale *Learning Healthcare System*.

## Modularity and Economics

My students and I have contributed to the understanding of modularity, economics, and evolvability in numerous ways. We contributed the fundamental notion of crosscutting interfaces (XPI) to aspect-oriented programming and software design. We were first to employ design structure matrices to model and analyze software modularity. We were also first to use real options to understand the economic value of software modularity.

## RECENT RESEARCH DEVELOPMENTS

- Spacemaker is a web-based, model-driven engineering tool that uses relational logic model finders to generate Pareto-optimal relational database schemas from object-oriented software models.
- Docility is a prototype web-based software tool for the collaborative development of definitions of key non-functional systems properties and for the statement and support of propositions about tradeoffs among them.

## RECENT GRANTS

- SIT/SERC-Ilities Tradespace and Affordability Project
- NSF-EAGER:Software Engineering Research for Societal Grand Challenge Problems
- SIT/SERC-A Cyber-Social Systems Approach to Making, Modeling and Managing Contingency Bases

**SEAS Research Information**
Pamela M. Norris,
Executive Associate Dean for Research
University of Virginia
Box 400232
Charlottesville, VA 22903
EngineeringResearch@virginia.edu
434.243.7683

University of Virginia | School of Engineering & Applied Science