



Educating the Next Generation of Software Engineers *(and beyond)*



Richard Turner
Stevens Institute of Technology
Systems and Software Conference
April 2009

rtturner@stevens.edu



Agenda

- How the world has changed
- The current state of software engineering education
- Creating and disseminating a new reference curriculum
- And next?



There are precious few interesting man-made systems whose success is not critically dependent on software.

Twenty years from now, software people will be sitting at the table and the other disciplines will be sitting around the sides of the room.

Eberhardt Rechtin, 1993

There are precious few interesting software systems anywhere whose success is not critically dependent on the developers practicing good systems engineering.



What do we teach for a master's degree in software engineering?

- The last effort to create a reference curriculum for graduate software engineering education was by the SEI in the early 1990s.
- There are, in effect, no current community-endorsed recommendations on what to teach software engineers – nothing that recognizes how the world has changed.
- Response: create a project to create a new reference curriculum in software engineering



The Integrated Software and Systems Engineering Curriculum Project

- Begun in May 2007 at Stevens Institute of Technology
- Sponsored by DoD Director of Systems and Software Engineering
- Three products planned:
 1. A modern reference curriculum for a master's degree in software engineering that integrates an appropriate amount of systems engineering
 2. A modern reference curriculum for a master's degree in systems engineering that integrates an appropriate amount of software engineering
 3. A truly interdisciplinary degree that is neither systems nor software engineering – it is both

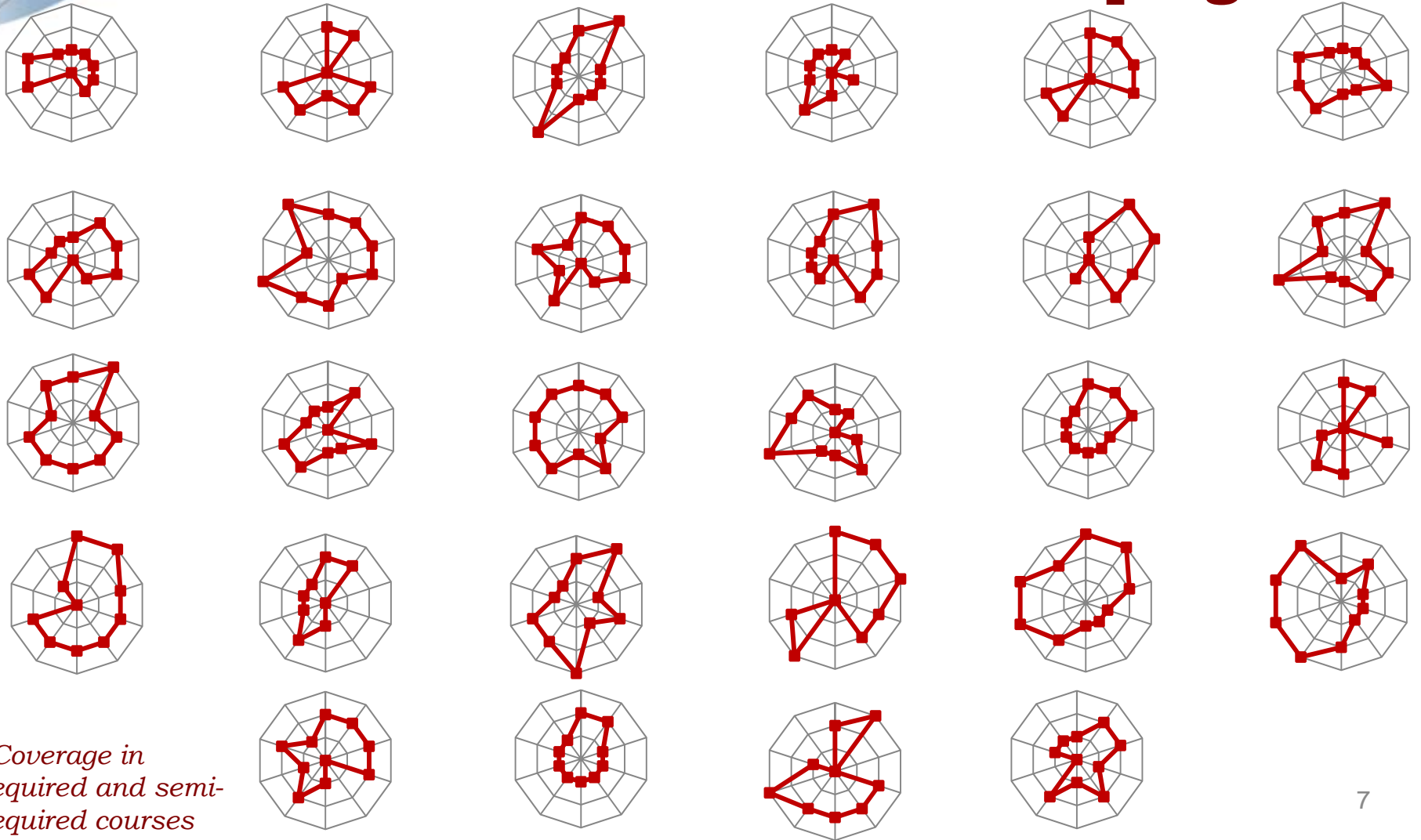


1st Project – Graduate Software Engineering Reference Curriculum

1. Understand the current state of SwE graduate education (November 2007)
2. Create GSwERC 0.25 with a small team, suitable for limited review (February 2008)
3. Publicize effort through conferences, papers, website, etc (continuous)
4. Create GSwERC 0.50 suitable for broad community review and early adoption (October 2008)
5. Create GSwERC 1.0 suitable for broad adoption (2009)
6. Transition stewardship to professional societies (2009)
7. Foster adoption world-wide (2010 and beyond)



SWEBOK coverage* across 28 SwE MS programs



**Coverage in required and semi-required courses*



The evolving author team

- Rick Adcock, *Cranfield University and INCOSE*
- Edward Alef, *General Motors*
- Mark Ardis, *Rochester Institute of Technology*
- Larry Bernstein, *Stevens Institute of Technology*
- Barry Boehm, *University of Southern California*
- Pierre Bourque, *Quebec University and SWEBOK volunteer*
- John Bracket, *Boston University*
- Murray Cantor, *IBM*
- Lillian Cassel, *Villanova and ACM participant*
- Robert Edson, *ANSER*
- Richard Fairley, *Colorado Technical University*
- Dennis Frailey, *Raytheon & Southern Methodist University*
- Gary Hafen, *Lockheed Martin and NDIA*
- Thomas Hilburn, *Embry-Riddle Aeronautical University*
- Greg Hislop, *Drexel University and IEEE Computer Society participant*
- Dave Klappholz, *Stevens Institute of Technology*
- Philippe Kruchten, *University of British Columbia*
- Phil Laplante, *Pennsylvania State University, Great Valley*
- Scott Lucero, *Department of Defense*
- Qiaoyun (Liz) Li, *Wuhan University, China*
- James McDonald, *Monmouth University*
- John McDermid, *University of York, UK*
- Ernest McDuffie, *National Coordination Office for NITRD*
- Bret Michael, *Naval Postgraduate School*
- William Milam, *Ford*
- Ken Nidiffer, *Software Engineering Institute*
- Art Pyster, *Stevens Institute of Technology*
- Doug Schmidt, *Vanderbilt University*
- Mary Shaw, *Carnegie Mellon University*
- Robert Suritis, *IBM*
- Richard Thayer, *California State University at Sacramento*
- Barrie Thompson, *Sunderland University, UK*
- Richard Turner, *Stevens Institute of Technology*
- Joseph Urban, *Texas Technical University*
- Ricardo Valerdi, *MIT & INCOSE*
- David Weiss, *Avaya*
- Mary Jane Willshire, *Colorado Technical University*



Creating GSwERC 0.50 and 1.0

1. Understand the current state of SWE graduate education (November 2007)
2. Create GSwERC 0.25 with a small team, suitable for limited review (February 2008)
3. Publicize effort through conferences, papers, website, etc (continuous)
4. *Create GSwERC 0.50 suitable for broad community review and early adoption (October 2008)*
5. *Create GSwERC 1.0 suitable for broad adoption (2009)*
6. Transition stewardship to professional societies (2009)
7. Foster adoption world-wide (2010 and beyond)



Expectations at entry

1. The equivalent of an undergraduate degree in computing or an undergraduate degree in an engineering or scientific field and a minor in computing
2. The equivalent of an introductory course in software engineering
3. At least two years of practical experience in some aspect of software engineering or software development



Outcomes 1 to 4 at graduation

1. Mastered the Core Body of Knowledge
2. Mastered at least one application domain, such as finance, medical, transportation, or telecommunications, and one application type, such as real-time, embedded, safety-critical, or highly distributed systems. That mastery includes understanding how differences in domain and type manifest themselves in both the software itself and in their engineering, and includes understanding how to learn a new application domain or type.
3. Mastered at least one knowledge area or sub-area from the Core Body of Knowledge to at least the Bloom Synthesis level.
4. Demonstrated how to make ethical professional decisions and practice ethical professional behavior.



Outcomes 5 to 7 at graduation

5. Understand the relationship between software engineering and systems engineering and be able to apply systems engineering principles and practices in the engineering of software.
6. Be able to work effectively as part of a team, including teams that may be international and geographically distributed, to develop quality software artifacts, and to lead in one area of project development, such as project management, requirements analysis, architecture, construction, or quality assurance.
7. Show ability to reconcile conflicting project objectives, finding acceptable compromises within limitations of cost, time, knowledge, existing systems, and organizations.

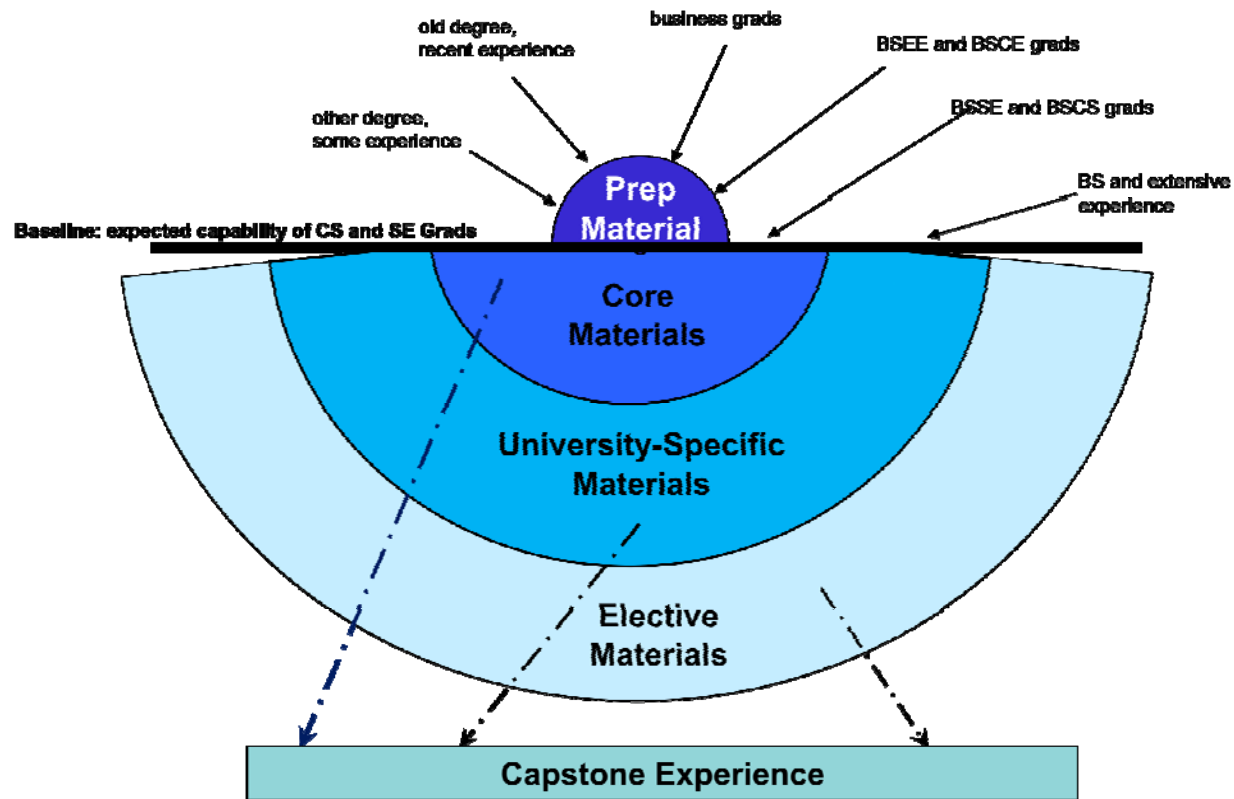


Outcomes 8 to 10 at graduation

8. Understand and appreciate the importance of feasibility analysis, negotiation, effective work habits, leadership, and good communication with stakeholders in a typical software development environment.
9. Understand how to learn new models, techniques, and technologies as they emerge, and appreciate the necessity of such continuing professional development.
10. Be able to analyze a current significant software technology, articulate its strengths and weaknesses, and specify and promote improvements or extensions to that technology.



Curriculum architecture





Implementation help

- Comparison of existing graduate software engineering programs with GSwERC recommendations – know how big the gap is between recommendations and practice
- Strategies recommended by the authors to implement GSwERC
- Hypothetical modifications of existing programs to more fully satisfy GSwERC

Seeking more involvement

SEEKING REVIEWERS AND EARLY ADOPTERS FOR VERSION 0.5



A New Reference Curriculum for Graduate Studies Leading to a Master's Degree in Software Engineering

Since August 2007, a group of over three 300 professionals from academia, industry, and government have been developing a new reference curriculum leading to a Master's Degree in Software Engineering. The new reference curriculum will integrate systems engineering into the education of software engineers and reflect the dramatic changes in how software is used and developed since the last major graduate reference curriculum was published in the early 1980's. The effort is endorsed by the International Council on Systems Engineering (INCISE), and the U.S. National Defense Industrial Association (NDIA) Systems Engineering Division. The IEEE Computer Society Educational Activities Board has a participating author and the ACM has a volunteer contributor. Sponsorship and funding for this effort are being provided by the U.S. Department of Defense.

Version 0.5 of the Graduate Software Engineering Reference Curriculum (GSwERC) was released on October 31, 2008 for broad review and early adoption. The document is posted on the GSwERC website (www.GSwERC.org). Review comments from all interested professionals are being sought. Instructions to provide a review are also provided on the GSwERC website. Version 1.0 is expected sometime in 2009.

One of the novel features of GSwERC is the inclusion of explicit comparisons of existing graduate software engineering programs to GSwERC recommendations and the inclusion of hypothetical modifications to two of those programs to better match GSwERC. These comparisons and modifications offer a window on how well GSwERC aligns with existing practice and will help faculty understand how to adopt GSwERC in their own universities. We welcome additional comparisons and hypothetical modifications from other universities to provide more insight into the gap between GSwERC and current practice and how to close that gap.

The Graduate Software Engineering Reference Curriculum (GSwERC) is ready for early adoption. Several of the current authors are now integrating portions of GSwERC into their own programs. The GSwERC author team is also interested in helping universities adopt GSwERC.

Please visit our website at
www.GSwERC.org
for more information.

Dr. Art Pyster, Distinguished Research
Professor, from Stevens Institute of
Technology in the United States, is the
project leader for GSwERC.

Please contact him at
art.pyster@stevens.edu if you are
interested in participating.

STEVENS
Institute of Technology

ASysT
INSTITUTE

- Requests sent off to about 2000 named reviewers + postings to various websites, etc.
- Reviews due March 15
- CSEET Workshop on February 18, proposed workshops at ITiCSE in July, panel at INCOSE IS, ...
- Want more comparisons + hypothetical implementations from across many countries

www.GSwERC.org



Issues remaining

1. What will post-version 1.0 governance be?
2. How will DAU, AFIT, and NPS adopt GSwERC?
3. Will OSD support the development of an SE curriculum which has the same rigor and reach as GSwERC?



Post Version 1.0 governance

- At January workshop, author team restructured GSwERC from a single very long volume into 3 volumes:
 - Vol I: Primary curriculum recommendations – heart of GSwERC
 - Vol II: Implementation guidance organized by specific programs who are compared with GSwERC, propose adopting GSwERC, or have experience adopting GSwERC
 - Vol III: Implementation guidance organized by issue, such as how to recruit faculty with the right skills, or how to organize projects with significant distributed development
- Volume I has content that professional societies traditionally shepherd
- Volumes II and III will provide insight from specific named programs and is not typically what professional societies shepherd



Post-version 1.0 governance model

- Must ensure stable governance post-version 1.0 release and strong adoption support.
- Can seek IEEE, ACM, and INCOSE joint governance of Volume I. Need to establish our expectations for what they will do with it once they control it. Expect updates around every 4 years. INCOSE likely to participate. IEEE reasonably likely to participate. ACM less clear.
- Another body could govern Volumes II and III. Frequent updates based on adoption experiences. Could include discussion forums, wikis, and other open collaboration structure more than “traditional” documents. Modest funding required for this. Governance by university collaboration led by Stevens or another university is likely at this point.
- Implementation workshops at conferences, summer faculty workshops, and other activities would promote adoption. Could seek NSF or other funding for this.



SE reference curriculum

- There is an INCOSE-sponsored SE reference curriculum, which was developed by a team led by Stevens.
- Unlike GSwERC, the SE curriculum development process was relatively modest effort without significant funding or widespread participation, doesn't have significant software engineering integration, and is not based on a robust and mature SE body of knowledge because none exists.
- INCOSE SE body of knowledge is relatively immature compared to the IEEE SwE body of knowledge and needs to be updated. (see <http://g2sebok.incose.org>)
- INCOSE would like to mature the SE body of knowledge, which would be a strong foundation on which to base an upgraded SE curriculum.
- OSD is considering a project for the SERC to update and mature SE body of knowledge and create mature SE reference curriculum. The effort would be similar to GSwERC with open collaborative international participation, fully shared resulting IP, with INCOSE, IEEE, and NDIA participation.



Summary

- GSwERC is on-track to deliver a fresh reference curriculum that may change how the world teaches graduate software engineering.
- Uncertainties remain in post-version 1.0 governance, but we have viable approach for way forward.
- There is a need and an opportunity for an SE project which could have similar impact.