

Development of SEBoK v. 2.7

Development of SEBoK v. 2.7

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

This version of the SEBoK was released 31 October 2022. This release included:

- A new article around loss-driven systems engineering, which specifically looks at system quality attributes like safety, security, and resilience through the lens of loss prevention.
- A new article on the history of systems engineering, which provides background on the early SE work leading up to more recent trends.
- Updates to the article on systems and industrial engineering.
- Minor updates to improve resources and align with evolving practices throughout part 3, including in the articles around systems engineering standards.
- Minor updates to the articles in Part 5, particularly the addition of new resources.
- Improvements in the SEBoK wiki infrastructure.

Contents

Governing Board for version 2.7

Dedication

SEBoK v. 2.7 Editorial Board

Student Editor

Editor's Corner

Governing Board for version 2.7

The three SEBoK steward organizations - the International Council on Systems Engineering (INCOSE), the Institute of Electrical and Electronics Engineers

Systems Council (IEEE-SYSC), and Stevens Institute of Technology - provide the primary funding and resources needed to sustain and evolve the SEBoK and make it available as a free and open resource to all. The stewards appoint the BKCASE Governing Board to be their primary agents to oversee and guide the SEBoK. The stewards appoint the SEBoK Editor in Chief to manage the SEBoK and oversee the Editorial Board.

The BKCASE Governing Board included:

- Representing the **The International Council on Systems Engineering (INCOSE)**
 - Art Pyster (Governing Board Chair), Emma Sparks
- Representing the **Systems Engineering Research Center (SERC)**
 - Thomas McDermott, Cihan Dagli
- Representing the **IEEE Systems Council (IEEE-SYSC)**
 - Stephanie White, Bob Rassa

Past governors include Andy Chen, Richard Fairley, Kevin Forsberg, Paul Frenz, Richard Hilliard, John Keppler, Bill Miller, David Newbern, Ken Nidiffer, Dave Olwell, Massood Towhidnejad, Jon Wade, David Walden, and Courtney Wright. The governors would especially like to acknowledge Andy Chen and Rich Hilliard, IEEE Computer Society, who were instrumental in helping the governors to work within the IEEE CS structure and who supported the SEBoK transition to the IEEE Systems Council.

Dedication



Barry Boehm (1935-2022) was a pioneer of the field and a joy to work with. Barry was an author and editor on the SEBoK for over a decade.

This release coincides with the 10th anniversary of the release of SEBoK v. 1.0. We would like to dedicate this

edition to our friend and colleague, Barry Boehm. Barry was not only an exceptional software and systems engineer but a true gentleman. Barry was a key member of the team that kicked off work on SEBoK v. 0.25 in 2009. Sadly, Barry passed away in August of this year, but his contributions to the field - which include the spiral development model and COCOMO among many - will live on, as will our fond memories of working with him. Rest well, friend.

Read more about Barry's contributions to systems engineering from his colleagues at the University of Southern California (USC) and the Systems Engineering Research Center (SERC).

SEBoK v. 2.7 Editorial Board

The SEBoK Editorial Board is chaired by the *Editor in Chief*, who provides the strategic vision for the SEBoK. The EIC is supported by a group of Editors, each of whom are responsible for a specific aspect of the SEBoK. The EIC and Editorial Board are supported by the Managing Editor, who handles all day-to-day operations. The EIC, Managing Editor, and Editorial Board are supported by a Student Editor, Madeline Haas, whose hard work and dedication are greatly appreciated.

SEBoK Editor in Chief



Robert J. Cloutier

University of South Alabama

rcloutier@southalabama.edu

Responsible for the appointment of SEBoK Editors and for the strategic direction and overall quality and coherence of the SEBoK.

SEBoK Managing Editor



Nicole Hutchison
Systems Engineering Research Center
nicole.hutchison@stevens.edu
or emtnicole@gmail.com
Responsible for the the day-to-day operations of the SEBoK and supports the Editor in Chief.

Each Editor has his/her area(s) of responsibility, or shared responsibility, highlighted in the table below.

SEBoK Part 1: SEBoK Introduction

Lead Editor: Robert J. Cloutier
University of South Alabama
rcloutier@southalabama.edu

SEBoK Part 2: Foundations of Systems Engineering

Lead Editor: Gary Smith (UK)
Airbus and International Society for the System Sciences
gary.r.smith@airbus.com
Responsible for the System Science Foundations of System Engineering.

Assistant Editor: Dov Dori
Massachusetts Institute of Technology (USA) and Technion Israel Institute of Technology (Israel)
dori@mit.edu
Responsible for the Representing Systems with Models knowledge area

Assistant Editor: Duane Hybertson
MITRE (USA)
dhyberts@mitre.org
Jointly responsible for the Systems Fundamentals, Systems Science and Systems Thinking knowledge areas.

Assistant Editor: Peter Tuddenham
College of Exploration (USA)
Peter@coexploration.net

Assistant Editor: Cihan Dagli
Missouri University of Science & Technology (USA)
dagli@mst.edu
Responsible for the Systems Approach Applied to Engineered Systems knowledge areas.

SEBoK Part 3: Systems Engineering and Management

Lead Editor: Jeffrey Carter
JTConsulting
jtcarter.57@outlook.com

Assistant Editor: Phyllis Marbach
INCOSE LA [USA]
pnarbach@gmail.com

Assistant Editor:

Assistant Editor:

Assistant Editor:

Assistant Editor:

Assistant Editor:

SEBoK Part 4: Applications of Systems Engineering

Lead Editor: Tom McDermott

Systems Engineering Research Center (SERC)

tmcdermo@stevens.edu

Assistant Editor: Judith Dahmann

MITRE Corporation (USA)

jdahmann@mitre.org

Jointly responsible for

Product Systems

Engineering and Systems of

Systems (SoS) knowledge

areas.

Assistant Editor: Javier

Calvo-Amodio

Oregon State University

Javier.Calvo@oregonstate.e

du

Assistant Editor: Michael Henshaw

Loughborough University (UK)

M.J.d.Henshaw@lboro.ac.uk

Jointly responsible for Product Systems Engineering and

Systems of Systems (SoS) knowledge areas

SEBoK Part 5: Enabling Systems Engineering

Lead Editor: Nicole Hutchison

Systems Engineering Research Center

nicole.hutchison@stevens.edu

Assistant Editor: Emma

Sparks

Cranfield University

Jointly responsible for the

Enabling Individuals and

Enabling Teams knowledge

areas.

Assistant Editor: Rick Hefner

California Institute of

Technology

Rick.Hefner@ngc.com

Assistant Editor: Bernardo Delicado

INCOSE/Indra Sistemas

bernardo.delicado@incose.org

SEBoK Part 6: Related Disciplines

Lead Editor: Art Pyster

George Mason University (USA)

apyster@gmu.edu

SEBoK Part 7: Systems Engineering Implementation Examples

Lead Editor: Clif Baldwin

FAA Technical Center

cliftonbaldwin@gmail.com

SEBoK Part 8: Emerging Knowledge

Lead Editor: Daniel DeLaurentis

Purdue University

ddelaure@purdue.edu

Assistant Editor: Ha Phuong Le
Purdue University
le135@purdue.edu

Student Editor

Madeline Haas, a student at George Mason University, is currently supporting the SEBoK and we gratefully acknowledge her exemplary efforts. Ms. Haas has also taken responsibility for managing the Emerging Research knowledge area of the SEBoK. The EIC and Managing Editor are very proud of the work Madeline has done and look forward to continuing to work with her.

Editor's Corner

The Editor's Corner is a space for the Editor in Chief to provide insights or perspective on relevant topics. The below is an archived copy of the article written by v. 2.5 EIC, Dr. Rob Cloutier.



30 October 2022

Let me point out before you begin reading, the purpose of this missive is not to point fingers, but rather to muse about whether we as a systems engineering community go back and reflect on previous prognostications. I feel it is important for individuals and organizations to reflect on the past, and see if we can learn anything to help us move forward as professionals. So, here goes ...

Have we developed a "resistance to change" - have we as a systems engineering community become slow and lethargic in responding to advances in tools and methodologies? A quick scan of the Internet shows that INCOSE has produced three vision documents since

2007. These vision documents represented the systems engineering vision for the years 2020, 2025, and the latest, for 2035. So let me ask - as a community, do we go back and look at those projections, determine what was correct, and what missed the mark? Do companies that support professional societies, or government agencies, go back and review these?

It is interesting that in 2007, INCOSE published a document titled Systems Engineering Vision 2020 (INCOSE-TP-2004-004-02). In the Executive Summary, it was stated that "In many respects, the future of systems engineering can be said to be model-based."

As I write this in September 2022, that original vision document was published 15 years ago. Yet, as the program director of a medium-sized systems engineering graduate program, I am approached regularly by companies, government agencies, and others looking for systems engineering talent. The number one skill these representatives are seeking is experience with model-based systems engineering (MBSE). The second ask is some exposure to modeling and simulation at the systems level.

In the INCOSE *Vision 2025*, released in 2014, one of the imperatives was cited as "Advancing the **tools** and **methods** to address complexity." (Emphasis added.) Today we are only 2+ years away from 2025. If I look at that vision document, and cherry-pick some of the statements from that 2025 vision around modeling, the authors of that document believed that by 2025:

- "Modeling and simulation is widely used to support integrated planning for a better representation of real-world constraints and solutions"
- "Systems engineering tools will facilitate systems engineering practices as part of a fully integrated engineering environment. Systems engineering tools will support high fidelity simulation, immersive technologies to support data visualization, semantic web technologies to support data integration, search, and reasoning, and communication technologies to support collaboration."
- "Model-based approaches will move

engineering and management from paper documentation as a communications medium to a paperless environment, by permitting the capture and review of systems design and performance in digital form."

Last week, the University hosted a job fair. As I walked around talking to DoD and industry talent managers, every company hiring systems engineers was begging for graduates that are familiar with MBSE, SysML, modeling, and simulation. It is obvious there is an overwhelming need that we, as academics, may not be satisfying. One student, a mechanical engineer, that took my Fundamentals of SE course, and then applied SysML in his Aerospace Engineering Master's thesis, was hired within 20 minutes of talking to a DoD agency. So, the prognostication was correct, but it seems there are still not enough individuals with the necessary skills. There seems to be an opportunity here - our systems engineering graduate programs need to redouble the efforts in these areas. Additionally, have our companies and agencies fall short of providing the requisite training? Is it realistic to believe systems engineers will learn these skills on their own?

Another interesting prediction in that vision document stated that, "Use of technologies such as simulation, visualization, and gaming will lead to innovations in systems engineering education." This is consistent with the Editor's Corner musing from the last two releases of the SEBoK that addressed a/the metaverse.

As I look forward to retirement, allow me to reflect on my time as a young engineer working on the V-22 Osprey in the mid 80's. The Boeing Corporation had decided to move from paper drawings to CAD/CAM. They tried to convince the aerospace engineers to give up their drafting board and tools and move to the computer based modeling tool called CATIA. The response from those aero engineers was "you can have my drafting tools when you pull them from my cold, dead fingers". So, Boeing hired a large number of college graduate engineers and trained them in CATIA. Drawings would come off the senior design engineers drafting boards, and these new CAD/CAM engineers would recreate the drawings in CATIA. Those digital models were then available to the numerical control machines on the factory floor. They were "resistance to change". Fast forward to today - the only place you will find a drafting board in Boeing is in the Boeing Museum in Seattle. All

of the senior engineers use CATIA (because they were the junior engineers in the 1980's. It took a generation or two to transition from paper-based to computer driven modeling.

Many of you reading this witnessed a similar story relative to the software engineering community. Even into the 90's, mainframe and Unix programming was performed using Vi and EMACS, or even using command line. Then, Borland introduced Turbo Pascal and their IDE (Integrated Development Environment) in 1983. Using this new IDE, programmers could program, compile, link, and execute a program all in one environment. For a decade or more, these IDE's were considered toys or crutches, and "real programmers" used Vi (or EMACS or command line, you get the idea). Again, an example of "resistance to change" by both the software engineering community, and the companies that would not pay for these new tools. Today, most software is written using IDE's... again, it took a generation or two.

I began this article stating that maybe the systems engineering community has become too lethargic and resistant to change - much like the aero engineers and software engineer were in the 80's/90's. Let's reflect on that claim. In 2007 it was identified by the INCOSE TechOps community that by 2020 our discipline would be "model-based". Yet, today, industry cannot find enough systems engineers with experience in model-based systems engineering. We had 15 years to prepare. Let's reflect on the other projection from that 2007 vision document that was identified earlier- "Use of technologies such as simulation, visualization, and gaming will lead to innovations in systems engineering education." This brings me back to the previous two Editor Corners - the metaverse. Has anyone in systems engineering taken this seriously and identified the requirements for such a system?

I feel that maybe this article has seemed like a random walk. So, what is really my destination? How do I wrap this up? If we are to look at the current systems engineering graduate education, I have been told that there are very few graduate SE programs right now that have integrated MBSE, digital engineering, and/or digital twin concepts woven into their curriculum. I am not talking about courses dedicated to these topics, but these topics simply integrated into the curriculum. In those programs that have done this, their students are being swept up months before they graduate with near six figure offers because they have used MBSE in several

classes and maybe on a research project, not to mention their thesis or dissertation.

Many hundreds of hours went into the creation of these Vision documents by leaders in our field from all over the world. It is my opinion that we should take more time to consider, reflect, and act on those vision documents. We should also take a look at them retrospectively and determine where they were right, and where they missed the mark. Then, we need to ask ourselves as a community, what do we need to do moving forward.

A handwritten signature in blue ink that reads "R. J. Cloutier". The signature is written in a cursive style with a large initial "R" and "J".

SEBoK v. 2.10, released 06 May 2024

Retrieved from

"https://sandbox.sebokwiki.org/index.php?title=Development_of_SEBoK_v._2.7&oldid=71140"

This page was last edited on 2 May 2024, at 21:54.