INCOSE Systems Engineering HANDBOOK

SYSTEMS ENGINEERING HANDBOOK

A GUIDE FOR SYSTEM LIFE CYCLE PROCESSES AND ACTIVITIES

FIFTH EDITION

INCOSE *International Council on Systems Engineering*

WILEY

SYSTEMS ENGINEERING HANDBOOK

SYSTEMS ENGINEERING HANDBOOK

A GUIDE FOR SYSTEM LIFE CYCLE PROCESSES AND ACTIVITIES

FIFTH EDITION

INCOSE-TP-2003–002-05
2023

Prepared by:

International Council on Systems Engineering (INCOSE)
7670 Opportunity Rd, Suite 220
San Diego, CA, USA 92111-2222

Compiled and Edited by:

DAVID D. WALDEN, ESEP — EDITOR-IN-CHIEF — AMERICAS SECTOR
THOMAS M. SHORTELL, CSEP — DEPUTY EDITOR-IN-CHIEF — AMERICAS SECTOR
GARRY J. ROEDLER, ESEP — EDITOR — AMERICAS SECTOR
BERNARDO A. DELICADO, ESEP — EDITOR — EMEA SECTOR
ODILE MORNAS, ESEP — EDITOR — EMEA SECTOR
YIP YEW-SENG, CSEP — EDITOR — ASIA OCEANIA SECTOR
DAVID ENDLER, ESEP — EDITOR — EMEA SECTOR

WILEY

This edition first published 2023
© 2023 John Wiley & Sons Ltd.

Edition History
Fourth edition, 2015

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by law. Advice on how to obtain permission to reuse material from this title is available at http://www.wiley.com/go/permissions.

The right of INCOSE; David Walden to be identified as the editorial material in this work has been asserted in accordance with law.

Registered Offices
John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, USA
John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, UK

For details of our global editorial offices, customer services, and more information about Wiley products visit us at www.wiley.com.

Wiley also publishes its books in a variety of electronic formats and by print-on-demand. Some content that appears in standard print versions of this book may not be available in other formats.

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

Limit of Liability/Disclaimer of Warranty

While the publisher and authors have used their best efforts in preparing this work, they make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives, written sales materials or promotional statements for this work. This work is sold with the understanding that the publisher is not engaged in rendering professional services. The advice and strategies contained herein may not be suitable for your situation. You should consult with a specialist where appropriate. The fact that an organization, website, or product is referred to in this work as a citation and/or potential source of further information does not mean that the publisher and authors endorse the information or services the organization, website, or product may provide or recommendations it may make. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

Library of Congress Cataloging-in-Publication Data

Names: Walden, David D., editor. | International Council on Systems Engineering, editor.
Title: INCOSE systems engineering handbook / edited by INCOSE, David Walden.
Description: Fifth edition. | Hoboken, NJ : John Wiley & Sons Ltd., [2023] | Includes index.
Identifiers: LCCN 2023022915 | ISBN 9781119814290 (paperback) | ISBN 9781119814306 (adobe pdf) | ISBN 9781119814313 (epub)
Subjects: LCSH: Systems engineering--Handbooks, manuals, etc. | Product life cycle--Handbooks, manuals, etc.
Classification: LCC TA168 .I444 2023 | DDC 620/.0042--dc23/eng/20230525
LC record available at https://lccn.loc.gov/2023022915

Cover Design: Wiley
Cover Images: © DANNY HU/Getty Images, © Zero Creatives/Getty Images, © Stebenkov Roman/Shutterstock, © Phonlamai Photo/Shutterstock, © MNBB Studio/Shutterstock, © Titima Ongkantong/Shutterstock

Set in 10/12pt TimesLTStd by Integra Software Services Pvt. Ltd., Pondicherry, India

CONTENTS

INCOSE NOTICES

This International Council on Systems Engineering (INCOSE) Technical Product was prepared by the INCOSE Systems Engineering Handbook Team. It is approved by the INCOSE Technical Operations Leadership for release as an INCOSE Technical Product.

Copyright ©2023 by INCOSE, subject to the following restrictions:

Author Use: Authors have full rights to use their contributions in a totally unfettered way with credit to the INCOSE technical source, except as noted in the following text. Abstraction is permitted with credit to the source.

INCOSE Use: Permission to reproduce and use this document or parts thereof by members of INCOSE and to prepare derivative works from this document for INCOSE use is granted, with attribution to INCOSE and the original author(s) where practical, provided this copyright notice is included with all reproductions and derivative works. Content from ISO/IEC/IEEE 15288 and ISO/IEC TR 24748-1 is used by permission and is not to be reproduced other than as part of this total document.

External Use: This document may not be shared or distributed to any non-INCOSE third party. Requests for permission to reproduce this document in whole or in part, or to prepare derivative works of this document for external and/or commercial use should be addressed to the INCOSE Central Office, 7670 Opportunity Road, Suite 220, San Diego, CA 92111-2222, USA.

Electronic Version Use: Any electronic version of this document is to be used for personal, professional use only and is not to be placed on a non-INCOSE sponsored server for general use.

Any additional use of these materials must have written approval from the INCOSE Central.

General Citation Guidelines: References to this handbook should be formatted as follows, with appropriate adjustments for formally recognized styles:

INCOSE SEH (2023). *Systems Engineering Handbook: A Guide for System Life Cycle Process and Activities* (5th ed.). D. D. Walden, T. M. Shortell, G. J. Roedler, B. A. Delicado, O. Mornas, Yip Y. S., and D. Endler (Eds.). San Diego, CA: International Council on Systems Engineering. Published by John Wiley & Sons, Inc.

HISTORY OF CHANGES

Revision	Revision date	Change description and rationale
Original	Jun 1994	Draft *Systems Engineering Handbook* (SEH) created by INCOSE members from several defense/aerospace companies—including Lockheed, TRW, Northrop Grumman, Ford Aerospace, and the Center for Systems Management—for INCOSE review.
1.0	Jan 1998	Initial SEH release approved to update and broaden coverage of SE process. Included broad participation of INCOSE members as authors. Based on Interim Standards EIA 632 and IEEE 1220.
2.0	Jul 2000	Expanded coverage on several topics, such as functional analysis. This version was the basis for the development of the Certified Systems Engineering Professional (CSEP) exam.
2.0A	Jun 2004	Reduced page count of SEH v2 by 25% and reduced the US DoD-centric material wherever possible. This version was the basis for the first publicly offered CSEP exam.
3.0	Jun 2006	Significant revision based on ISO/IEC 15288:2002. The intent was to create a country- and domain-neutral handbook. Significantly reduced the page count, with elaboration to be provided in appendices posted online in the INCOSE Product Asset Library (IPAL).
3.1	Aug 2007	Added detail that was not included in SEH v3, mainly in new appendices. This version was the basis for the updated CSEP exam.
3.2	Jan 2010	Updated version based on ISO/IEC/IEEE 15288:2008. Significant restructuring of the handbook to consolidate related topics.
3.2.1	Jan 2011	Clarified definition material, architectural frameworks, concept of operations references, risk references, and editorial corrections based on ISO/IEC review.
3.2.2	Oct 2011	Correction of errata introduced by revision 3.2.1.
4.0	Jul 2015	Significant revision based on ISO/IEC/IEEE 15288:2015, inputs from the relevant INCOSE working groups (WGs), and to be consistent with the Guide to the Systems Engineering Body of Knowledge (SEBoK).
5.0	Jul 2023	Significant revision based on ISO/IEC/IEEE 15288:2023 and inputs from the relevant INCOSE working groups (WGs). Significant restructuring of the handbook based inputs from INCOSE stakeholders.

LIST OF FIGURES

LIST OF TABLES

PREFACE

The objective of the International Council on Systems Engineering (INCOSE) *Systems Engineering Handbook* (SEH) is to describe key Systems Engineering (SE) process activities. The intended audience is the SE practitioner. When the term “SE practitioner” is used in this handbook, it includes the new SE practitioner, a product engineer, an engineer in another discipline who needs to perform SE, or an experienced SE practitioner who needs a convenient reference.

The descriptions in this handbook show what each SE process activity entails, in the context of designing for required performance and life cycle considerations. On some projects, a given activity may be performed very informally; on other projects, it may be performed very formally, with interim products under formal configuration control. This document is not intended to advocate any level of formality as necessary or appropriate in all situations. The appropriate degree of formality in the execution of any SE process activity is determined by the following:

The need for communication of what is being done (across members of a project team, across organizations, or over time to support future activities)
The level of uncertainty
The degree of complexity
The consequences to human welfare

On smaller projects, where the span of required communications is small (few people and short project life cycle) and the cost of rework is low, SE activities can be conducted very informally and thus at low cost. On larger projects, where the span of required communications is large (many teams that may span multiple geographic locations and organizations and long project life cycle) and the cost of failure or rework is high, increased formality can significantly help in achieving project opportunities and in mitigating project risk.

In a project environment, work necessary to accomplish project objectives is considered “in scope”; all other work is considered “out of scope.” On every project, “thinking” is always “in scope.” Thoughtful tailoring and intelligent application of the SE processes described in this handbook are essential to achieve the proper balance between the risk of missing project technical and business objectives on the one hand and process paralysis on the other hand. Part IV provides tailoring and application guidance to help achieve that balance.

APPROVED FOR THE INCOSE SEH FIFTH EDITION:

Christopher D. Hoffman, CSEP, INCOSE Technical Director, January 2021-January 2023
Olivier Dessoude, INCOSE Technical Director, January 2023-January 2025
Theodore J. Ferrell, INCOSE Assistant Director, Technical Review, January 2021-January 2023
Krystal Porter, INCOSE Assistant Director, Technical Review, January 2023-January 2025
Lori F. Zipes, ESEP, INCOSE Assistant Director, Technical Information, January 2022-January 2024
Tony Williams, ESEP, INCOSE Assistant Director, Product Champion, January 2022-January 2025

HOW TO USE THIS HANDBOOK

PURPOSE

This handbook defines the “state-of-the-good-practice” for the discipline of Systems Engineering (SE) and provides an authoritative reference to understand the SE discipline in terms of content and practice.

APPLICATION

This handbook is consistent with ISO/IEC/IEEE 15288 (2023), *Systems and software engineering—System life cycle processes*, hereafter referred to as ISO/IEC/IEEE 15288, to ensure its usefulness across a wide range of application domains for engineered systems and products, as well as services. ISO/IEC/IEEE 15288 is an international standard that provides system life cycle process outcomes, activities, and tasks, whereas this handbook further elaborates on the activities and practices necessary to execute the processes.

This handbook is also consistent with the *Guide to the Systems Engineering Body of Knowledge*, hereafter referred to as the SEBoK (2023), to the extent practicable. In many places, this handbook points readers to the SEBoK for more detailed coverage of the related topics, including a current and vetted set of references. The SEBoK also includes coverage of “state-of-the-art” in SE.

For organizations that do not follow the principles of ISO/IEC/IEEE 15288 or the SEBoK to specify their life cycle processes, this handbook can serve as a reference to practices and methods that have proven beneficial to the SE community at large and that can add significant value in new domains, if appropriately selected, tailored, and applied. Part IV provides top-level guidance on the application of SE in selected product sectors and domains.

Before applying this handbook in a given organization or on a given project, it is recommended that the tailoring guidelines in Part IV be used to remove conflicts with existing policies, procedures, and standards already in use within an organization. Not every process will apply universally. Careful selection from the material is recommended. Reliance on process over progress will not deliver a system. Processes and activities in this handbook do not supersede any international, national, or local laws or regulations.

USAGE

This handbook was developed to support the users and use cases shown in Table 0.1. Primary users are those who will use the handbook directly. Secondary users are those who will typically use the handbook with assistance from SE practitioners. Other users and use cases are possible.

TABLE 0.1 Handbook users and use cases

User	Type	Use cases
Seasoned SE Practitioner. Those who need to reinforce, refresh, and renew their SE knowledge	Primary	• Adapt or refer to handbook to suit individual applicability • Explore good practices • Identify blind spots or gaps by providing a good checklist to ensure necessary coverage • References to other sources for more in-depth understanding
Novice SE Practitioner: Those who need to start using SE	Primary	• Support structured, coherent, and comprehensive learning • Understand the scope (breadth and depth) of systems thinking and SE practices
INCOSE Certification: Systems Engineering Professional (SEP) certifiers and those being certified	Primary	• Define body of knowledge for SEP certification • Form the basis of the SEP examination
SE Educators: Those who develop and teach SE courses, including universities and trainers	Primary	• Support structured, coherent, and comprehensive learning • Suggest relevant SE topics to trainers for their course content • Serve as a supplemental teaching aid
SE Tool Providers/Vendors: Those who provide tools and methods to support SE practitioners	Primary	• Suggest tools, methods, or other solutions to be developed that help practitioners in their work
Prospective SE Practitioner or Manager: Those who may be interested in pursuing a career in SE or who need to be aware of SE practices	Secondary	• Provide an entry level survey to understand what SE is about to someone who has a basic technical or engineering background
Interactors: Those who perform in disciplines that exchange (consume and/or produce) information with SE practitioners	Secondary	• Understand basic terminologies, scope, structure, and value of SE • Understand the role of the SE practitioner and their relationship to others in a project or an organization

INCOSE SEH original table created by Yip. Usage per the INCOSE Notices page. All other rights reserved.

ORGANIZATION AND STRUCTURE

As shown in Figure 0.1, this handbook is organized into six major parts, plus appendices.

Systems Engineering Introduction (Part I) provides foundational SE concepts and principles that underpin all other parts. It includes the what and why of SE and why it is important, key definitions, systems science and systems thinking, and SE principles and concepts.

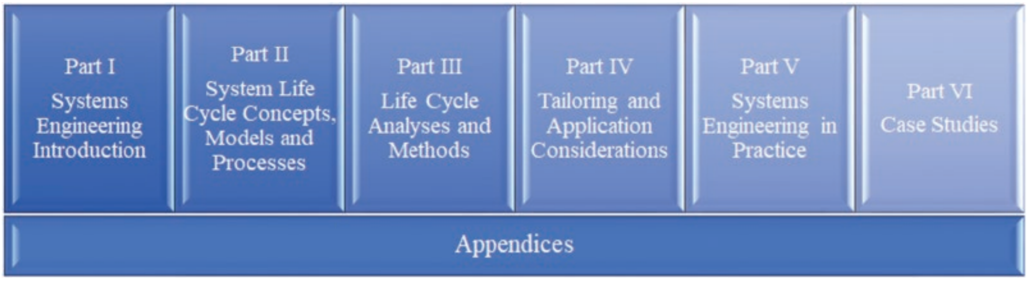

FIGURE 0.1 Handbook structure. INCOSE SEH original figure created by Mornas. Usage per the INCOSE Notices page. All other rights reserved.

System Life Cycle Concepts, Models, and Processes (Part II) describes an informative life cycle model with six stages: concept, development, production, utilization, support, and retirement. It also describes a set of life cycle processes to support SE consistent with the four process groups of ISO/IEC/IEEE 15288: Agreement Processes, Organizational Project Enabling Processes, Technical Management Processes, and Technical Processes.

Life Cycle Analyses and Methods (Part III) describes a set of quality characteristics approaches that need to be considered across the system life cycle. This part also describes methods that can apply across all processes, reflecting various aspects of the concurrent, iterative, and recursive nature of SE.

Tailoring and Application Considerations (Part IV) describes information on how to tailor (adapt and scale) the SE processes. It also introduces various considerations to view and apply SE: SE methodologies and approaches, system types, and project sectors and domains.

Systems Engineering in Practice (Part V) describes SE competencies, diversity, equity, and inclusion, SE relationship to other disciplines, SE transformation, and insight into the future of SE.

Case Studies (Part VI) describes several case studies that are used throughout the handbook to reinforce the SE principles and concepts.

Appendix A contains a list of references used in this handbook. Appendices B and C provide a list of acronyms and a glossary of SE terms and definitions, respectively. Appendix D provides an N^2 diagram of the SE life cycle processes showing an example of the dependencies that exist in the form of shared inputs or outputs. Appendix E provides a list of all the typical inputs/outputs identified for each SE life cycle process. Appendix F acknowledges the various contributors to this handbook. Errors, omissions, and other suggestions for this handbook can be submitted to the INCOSE using instructions found in Appendix G.

SYMBOLOGY

As described in Section 2.3.1.2, SE is a concurrent, iterative, and recursive process. The following symbology is used throughout this handbook to reinforce these concepts

	*Concurrency* is indicated by the parallel lines. *Iteration* is indicated by the circular arrows.
	*Recursion* is indicated by the down and up arrows.

TERMINOLOGY

One of the SE practitioner's first and most important responsibilities on a project is to establish nomenclature and terminology that support clear, unambiguous communication and definition of the system and its elements, functions, operations, and associated processes. Further, to promote the advancement of the field of SE throughout the world, it is essential that common definitions and understandings be established regarding general methods and terminology that in turn support common processes. As more SE practitioners accept and use common terminology, SE will experience improvements in communications, understanding, and, ultimately, productivity.

The glossary of terms used throughout this book (see Appendix C) is based on the definitions found in ISO/IEC/IEEE 15288; ISO/IEC/IEEE 24765 (2017); and the SEBoK.

1

SYSTEMS ENGINEERING INTRODUCTION

1.1 WHAT IS SYSTEMS ENGINEERING?

Systems Engineering (SE)

Our world and the systems we engineer continue to become more complex and interrelated. SE is an integrative approach to help teams collaborate to understand and manage systems and their complexity and deliver successful systems. The SE perspective is based on systems thinking—a perspective that sharpens our awareness of wholes and how the parts within those wholes interrelate (incose.org, *About Systems Engineering*). SE aims to ensure the pieces work together to achieve the objectives of the whole. SE practitioners work within a project team and take a holistic, balanced, life cycle approach to support the successful completion of system projects (INCOSE Vision 2035, 2022). SE has the responsibility to realize systems that are *fit for purpose*, namely that systems accomplish their intended purposes and be resilient to effects in real-world operation, while minimizing unintended actions, side effects, and consequences (Griffin, 2010).

Definition of SE

INCOSE Definitions (2019) and ISO/IEC/IEEE 15288 (2023) define:

> **Systems Engineering** is a transdisciplinary and integrative approach to enable the successful realization, use, and retirement of engineered systems, using systems principles and concepts, and scientific, technological, and management methods.

INCOSE Definitions (2019) elaborates:
SE focuses on:

INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Fifth Edition.
Edited by David D. Walden, Thomas M. Shortell, Garry J. Roedler, Bernardo A. Delicado, Odile Mornas, Yip Yew-Seng, and David Endler.

- establishing, balancing and integrating stakeholders' goals, purpose and success criteria, and defining actual or anticipated stakeholder needs, operational concepts, and required functionality, starting early in the development cycle;
- establishing an appropriate life cycle model, process approach and governance structures, considering the levels of complexity, uncertainty, change, and variety;
- generating and evaluating alternative solution concepts and architectures;
- baselining and modeling requirements and selected solution architecture for each stage of the endeavor;
- performing design synthesis and system verification and validation;
- while considering both the problem and solution domains, taking into account necessary enabling systems and services, identifying the role that the parts and the relationships between the parts play with respect to the overall behavior and performance of the system, and determining how to balance all of these factors to achieve a satisfactory outcome.

SE provides facilitation, guidance, and leadership to integrate the relevant disciplines and specialty groups into a cohesive effort, forming an appropriately structured development process that proceeds from concept to development, production, utilization, support, and eventual retirement.

SE considers both the business and the technical needs of acquirers with the goal of providing a quality solution that meets the needs of users and other stakeholders, is fit for the intended purpose in real-world operation, and avoids or minimizes adverse unintended consequences.

The goal of all SE activities is to manage risk, including the risk of not delivering what the acquirer wants and needs, the risk of late delivery, the risk of excess cost, and the risk of negative unintended consequences. One measure of utility of SE activities is the degree to which such risk is reduced. Conversely, a measure of acceptability of absence of a SE activity is the level of excess risk incurred as a result.

Definitions of System

While the concepts of a *system* can generally be traced back to early Western philosophy and later to science, the concept most familiar to SE practitioners is often traced to Ludwig von Bertalanffy (1950, 1968) in which a system is regarded as a "whole" consisting of interacting "parts."

INCOSE Definitions (2019) and ISO/IEC/IEEE 15288 (2023) define:

> A **system** is an arrangement of parts or elements that together exhibit behavior or meaning that the individual constituents do not.

A system is sometimes considered as a product or as the services it provides.

In practice, the interpretation of its meaning is frequently clarified using an associative noun (e.g., medical system, aircraft system). Alternatively, the word "system" is substituted simply by a context-dependent synonym (e.g., pacemaker, aircraft), though this potentially obscures a system principles perspective.

A complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services, and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment.

INCOSE Definitions (2019) elaborates:
Systems can be either physical or conceptual, or a combination of both. Systems in the physical universe are composed of matter and energy, may embody information encoded in matter-energy carriers, and exhibit observable behavior. Conceptual systems are abstract systems of pure information, and do not directly exhibit behavior, but exhibit "meaning." In both cases, the system's properties (as a whole) result, or emerge, from:

a) the parts or elements and their individual properties,
b) the relationships and interactions between and among the parts, the system, other external systems (including humans), and the environment.

SE practitioners are especially interested in systems which have or will be “systems engineered” for a purpose. Therefore, INCOSE Definitions (2019) defines:

> An **engineered system** is a system designed or adapted to interact with an anticipated operational environment to achieve one or more intended purposes while complying with applicable constraints.

“Engineered systems” may be composed of any or all of the following elements: people, products, services, information, processes, and/or natural elements.

Origins and Evolution of SE

Aspects of SE have been applied to technical endeavors throughout history. However, SE has only been formalized as an engineering discipline beginning in the early to middle of the twentieth century (INCOSE Vision 2035, 2022). The term “systems engineering” dates to Bell Telephone Laboratories in the early 1940s (Fagen, 1978; Hall, 1962; Schlager, 1956). Fagen (1978) traces the concepts of SE within the Bell System back to early 1900s and describes major applications of SE during World War II. The British used multidisciplinary teams to analyze their air defense system in the 1930s (Martin, 1996). The RAND Corporation was founded in 1946 by the United States Air Force and claims to have created “systems analysis.” Hall (1962) asserts that the first attempt to teach SE as we know it today came in 1950 at MIT by Mr. Gilman, Director of Systems Engineering at Bell. TRW (now a part of Northrop Grumman) claims to have “invented” SE in the late 1950s to support work with ballistic missiles. Goode and Machol (1957) authored the first book on SE in 1957. In 1990, a professional society for SE, the National Council on Systems Engineering (NCOSE), was founded by representatives from several US corporations and organizations. As a result of growing involvement from SE practitioners outside of the US, the name of the organization was changed to the International Council on Systems Engineering (INCOSE) in 1995 (incose.org, *History of Systems Engineering*; Buede and Miller, 2016).

With the introduction of the international standard ISO/IEC 15288 in 2002, the discipline of SE was formally recognized as a preferred mechanism to establish agreement for the creation of products and services to be traded between two or more organizations—the supplier(s) and the acquirer(s). This handbook builds upon the concepts in the latest edition of ISO/IEC/IEEE 15288 (2023) by providing additional context, definitions, and practical applications. Table 1.1 provides a list of key SE standards and guides related to the content of this handbook.

TABLE 1.1 SE standards and guides

Reference	Title
ISO/IEC/IEEE 15026	Systems and software engineering—Systems and software assurance (Multi-part standard)
ISO/IEC/IEEE 15288	Systems and software engineering—System life cycle processes
IEEE/ISO/IEC 15289	Systems and software engineering—Content of life cycle information items (documentation)
ISO/IEC/IEEE 15939	Systems and software engineering—Measurement process
ISO/IEC/IEEE 16085	Systems and software engineering—Life cycle processes—Risk management
ISO/IEC/IEEE 16326	Systems and software engineering—Life cycle processes—Project management
ISO/IEC/IEEE 21839	Systems and software engineering—System of systems (SoS) considerations in life cycle stages of a system
ISO/IEC/IEEE 21840	Systems and software engineering—Guidelines for the utilization of ISO/IEC/IEEE 15288 in the context of system of systems (SoS)
ISO/IEC/IEEE 21841	Systems and software engineering—Taxonomy of systems of systems
ISO/IEC/IEEE 24641	Systems and software engineering—Methods and tools for model-based systems and software engineering

(Continued)

TABLE 1.1 (Continued)

Reference	Title
ISO/IEC/IEEE 24748–1	Systems and software engineering—Life cycle management—Part 1: Guidelines for life cycle management
ISO/IEC/IEEE 24748–2	Systems and software engineering—Life cycle management—Part 2: Guidelines for the application of ISO/IEC/IEEE 15288
ISO/IEC/IEEE 24748–4	Systems and software engineering—Life cycle management—Part 4: Systems engineering planning
ISO/IEC/IEEE 24748–6	Systems and software engineering—Life cycle management—Part 6: System integration engineering
ISO/IEC/IEEE 24748–7	Systems and software engineering—Life cycle management—Part 7: Application of systems engineering on defense programs
ISO/IEC/IEEE 24748–8 / IEEE 15288.2	Systems and software engineering—Life cycle management—Part 8: Technical reviews and audits on defense programs
ISO/IEC/IEEE 24765	Systems and software engineering—Vocabulary
ISO/IEC/IEEE 26550	Software and systems engineering—Reference model for product line engineering and management
ISO/IEC/IEEE 26580	Software and systems engineering—Methods and tools for the feature-based approach to software and systems product line engineering
ISO/IEC/IEEE 29148	Systems and software engineering—Life cycle processes—Requirements engineering
ISO/IEC/IEEE 42010	Systems and software engineering—Architecture description
ISO/IEC/IEEE 42020	Software, systems and enterprise—Architecture processes
ISO/IEC/IEEE 42030	Software, systems and enterprise—Architecture evaluation framework
ISO/IEC 29110	Systems and Software Engineering Standards and Guides for Very Small Entities (VSEs) (Multi-part set)
ISO/IEC 31000	Risk management
ISO/IEC 31010	Risk management—Risk assessment techniques
ISO/IEC 33060	Process assessment—Process assessment model for system life cycle processes
ISO/PAS 19450	Automation systems and integration—Object-Process Methodology (OPM)
ISO 10007	Quality management—Guidelines for configuration management
ISO 10303-233	Industrial automation systems and integration—Product data representation and exchange—Part 233: Application protocol: Systems engineering
NIST SP 800–160 Vol. 1	Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems
NIST SP 800–160 Vol. 2	Developing Cyber-Resilient Systems: A Systems Security Engineering Approach
OMG SysML™	OMG Systems Modeling Language
SEBoK	Guide to the Systems Engineering Body of Knowledge (SEBoK)
SAE-EIA 649C	Configuration Management Standard
SAE 1001	Integrated Project Processes for Engineering a System (Note: Replaced ANSI/EIA 632)
ANSI/AIA.A G.043B	Guide to the Preparation of Operational Concept Documents
CMMI	CMMI® V2.0

INCOSE SEH original table created by Mornas, Roedler, and Walden. Usage per the INCOSE Notices page. All other rights reserved.

1.2 WHY IS SYSTEMS ENGINEERING IMPORTANT?

The purpose of SE is to conceive, develop, produce, utilize, support, and retire the right product or service within budget and schedule constraints. Delivering the right product or service requires a common understanding of the current system state and a common vision of the system's future states, as well as a methodology to transform a set of stakeholder needs, expectations, and constraints into a solution. The right product or service is one that accomplishes the required service or mission. A common vision and understanding, shared by acquirers and suppliers, is achieved through application of proven methods that are based on standard approaches across people, processes, and tools. The application of these methods is continuous throughout the system's life cycle.

SE is particularly important in the presence of complexity (see Section 1.3.7). Most current systems are formed by integrating commercially available products or by integrating independently managed and operated systems to provide emergent capabilities which increase the level of complexity (see Sections 4.3.3 and 4.3.6). This increased reliance on off-the-shelf and systems of systems has significantly reduced the time from concept definition to market availability of products. Over the years between 1880 and 2000, average 25% market penetration has been reduced by more than a factor of four as illustrated in Figure 1.1.

In response to complexity and compressed timelines, SE methods and tools have become more adaptable and efficient. Introduction of agile methods (see Section 4.2.2) and SE modeling language standards such the Systems Modeling Language (SysML) have allowed SE practitioners to manage complexity and increase the implementation of a common system vision (see bottom of Figure 1.1). Model Based SE (MBSE) methods adoption continues to grow (see Section 4.2.1), particularly in the early conceptual design and requirements analysis (SEBOK, *Emerging Topics*). MBSE research literature continues to report on the increased productivity and quality of design and promises further progression toward a digital engineering (DE) approach, where data is transparent and cooperation optimized across all engineering disciplines. Standards organizations are updating or developing new approaches that take DE into consideration. SE will have to address this new digital representation of the system as DE becomes the way of doing business (see Section 5.4). The rapid evolution and introduction of Artificial Intelligence (AI) and Machine Learning (ML) into SE further increases complexity of verifiability, safety, and trust of self-learning and evolving systems.

The overall value of SE has been the subject of studies and papers from many organizations since the introduction of SE. A 2013 study was completed at the University of South Australia to quantify the return on investment (ROI) of SE activities on overall project cost and schedule (Honour, 2013). Figure 1.2 compares the total SE effort with cost

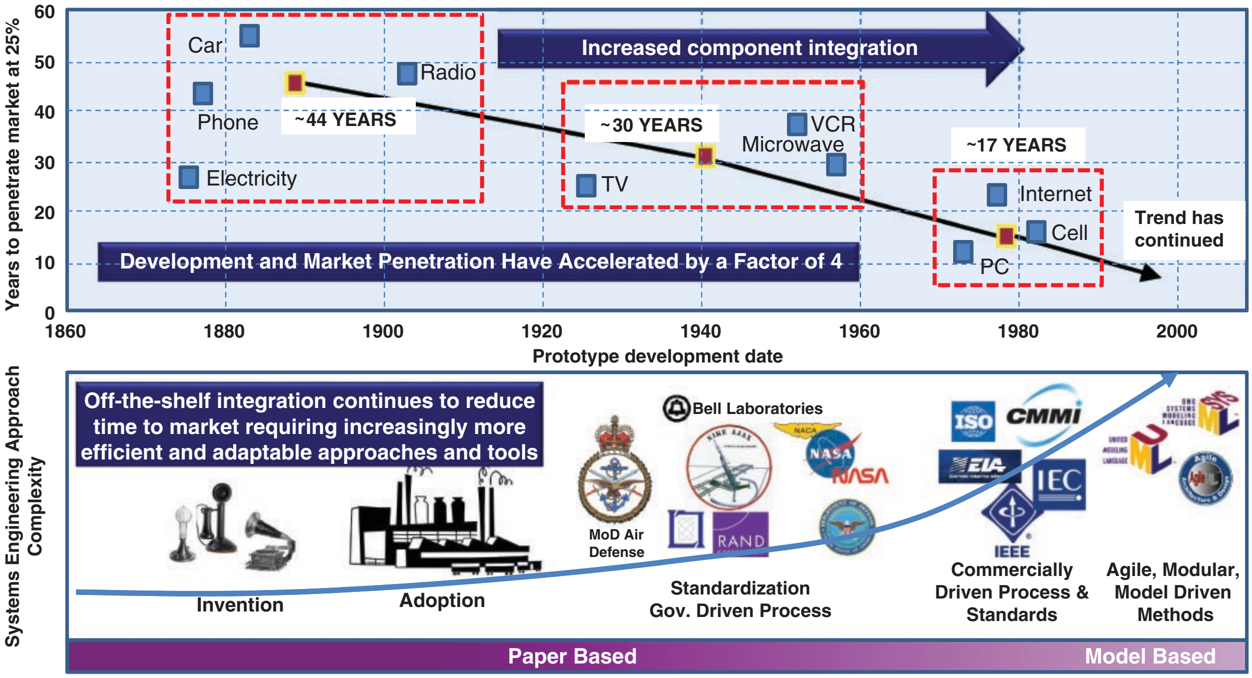

FIGURE 1.1 Acceleration of design to market life cycle has prompted development of more automated design methods and tools. INCOSE SEH original figure created by Amenabar. Usage per the INCOSE Notices page. All other rights reserved.

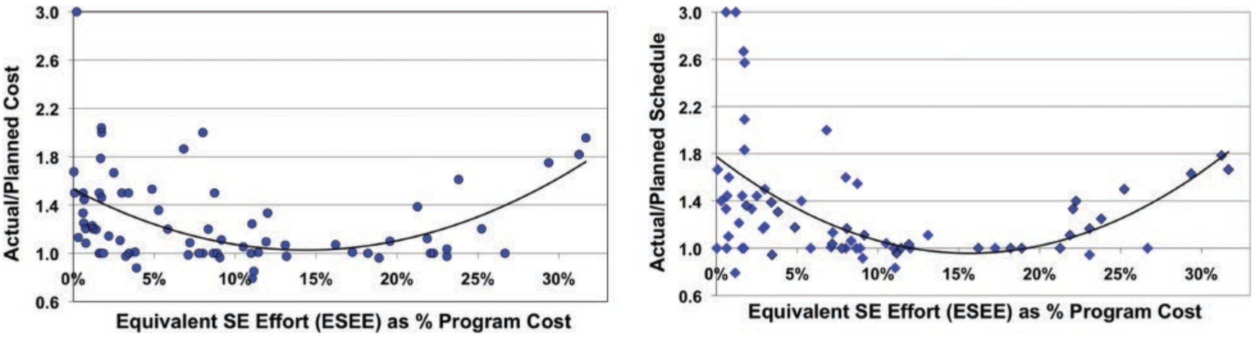

FIGURE 1.2 Cost and schedule overruns correlated with SE effort. From Honour (2013) with permission from University of South Wales. All other rights reserved.

compliance (left figure) and schedule performance (right figure). In both graphs, increasing the percentage of SE within the project results in better success up to an optimum level, above which SE ROI is diminished above those total program expenditure levels due to increased unwarranted processes. Study data shows that SE effort had a significant, quantifiable effect on project success, with correlation factors as high as 80%. Results show that the optimum level of SE effort for a normalized range of 10% to 14% of the total project cost.

The ROI of adding additional SE activities to a project is shown in Table 1.2, and it varies depending on the level of SE activities already in place. If the project is using no SE activities, then adding SE carries a 7:1 ROI; for each cost unit of additional SE, the project total cost will reduce by 7 cost units. At the median level of the projects interviewed, additional SE effort carries a 3.5:1 ROI.

A joint 2012 study by the National Defense Industrial Association (NDIA), the Institute of Electrical and Electronic Engineers (IEEE), and the Software Engineering Institute (SEI) of Carnegie Mellon University (CMU) surveyed 148 development projects and found clear and significant relationships between the application of SE activities and the performance of those projects as seen in Figure 1.3 (Elm and Goldenson, 2012). The study broke the projects by the maturity of their SE processes as measured by the quantity and quality of specific SE work products and considered the complexity of each project and the maturity of the technologies being implemented (n=number of projects). It also assessed the levels of project performance, as measured by satisfaction of budget, schedule, and technical requirements. The left column represents those projects deploying lower levels of SE expertise and capability. Among these projects, only 15% delivered higher levels of project performance and 52% delivered lower levels of project performance. The center column represents those projects deploying moderate levels of SE expertise and capability. Among these projects, the number delivering higher levels of project performance increased to 24% and those delivering lower levels decreased to 29%. The right column represents those projects deploying higher levels of SE expertise and capability. For these projects, the number delivering higher levels of project performance increased substantially

TABLE 1.2 SE return on investment

Current SE effort (% of program cost)	Average cost overrun (%)	ROI for additional SE effort (cost reduction $ per $ SE added)
0	53	7.0
5	24	4.6
7.2 (median of all programs)	15	3.5
10	7	2.1
15	3	–0.3
20	10	–2.8

From Honour (2013) with permission from University of South Wales. All other rights reserved.

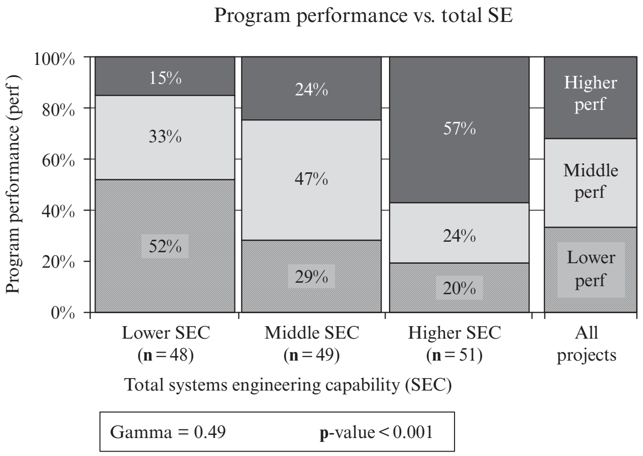

FIGURE 1.3 Project performance versus SE capability. From Elm and Goldenson (2012) with permission from Carnegie Mellon University. All other rights reserved.

to 57%, while those delivering lower levels decreased to 20%. As Figure 1.3 shows, well-applied SE increases the probability of successfully developing an engineered system.

A 1993 Defense Acquisition University (DAU) statistical analysis on US Department of Defense (DoD) projects examined spent and committed life cycle cost (LCC) over time (DAU, 1993). As illustrated notionally in Figure 1.4, an important result from this study is that by the time approximately 20% of the actual costs have been accrued, over 80% of the total LCC has already typically been committed. Figure 1.4 also shows that it is less costly to fix or address issues if they are identified early. Good SE practice is the means by which the issues are identified and ensures that the understanding obtained is applied as appropriate during the life cycle, thus reducing technical debt.

INCOSE maintains value proposition statements (INCOSE Value Strategic Initiative Report, 2021) as tailored to different areas and industries. Areas covered include individual INCOSE membership, organizational INCOSE membership, INCOSE SE certification, and the discipline of SE. Industries include commercial, government, and nonprofit organizations. A sample of these findings includes:

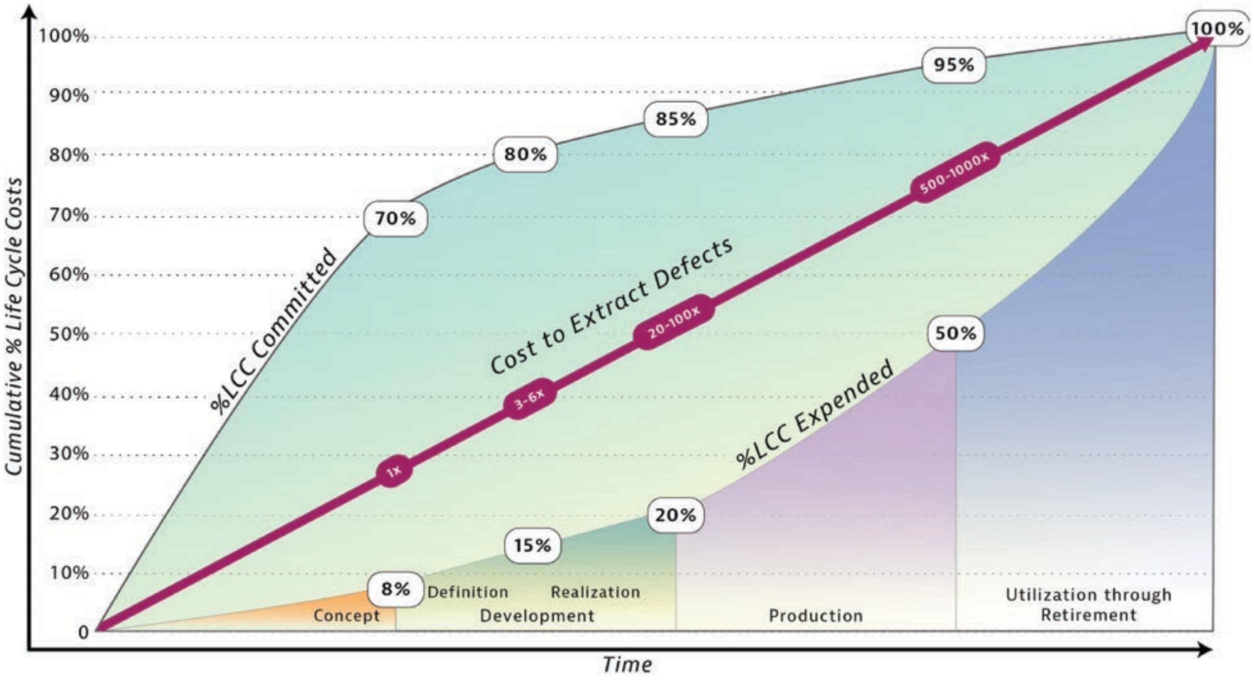

FIGURE 1.4 Life cycle costs and defect costs against time. INCOSE SEH original figure created by Walden derived from DAU (1993). Usage per the INCOSE Notices page. All other rights reserved.

- *Value of SE to the Commercial/Market-Driven Industry*: Companies and other enterprises in commercial industry will benefit from the internal practice of professional SE by having enhanced their capability for the development of innovative products and services for distribution in both mature and immature markets, in a more efficient and competitive manner.
- *Value of SE to Government/Infrastructure/Aerospace/Defense Industry*: SE provides a tailorable, systematic approach to all stages of a project, from concept to retirement. SE can accommodate different approaches including agile and sequential and facilitate commonality and open architectures to ensure lower acquisition, maintenance, and upgrade costs. By confirming correct and complete requirements and requirements allocations, the resulting design has fewer and less significant changes resulting in improved overall cost and schedule performance.
- *Value of SE to Nonprofit/Research Industry*: A nonprofit enterprise will benefit from the internal practice of professional SE by having enhanced their capability for the development of innovative client services in a more efficient and effective manner. An enterprise engaged in basic or applied research will benefit from the internal practice of SE by having enhanced its capabilities for discovery and invention that supports technology development in a more effective manner.

1.3 SYSTEMS CONCEPTS

Important system concepts include the system of interest (SoI), the system environment, and external systems. The boundaries between the system and the surrounding elements are important to understand. These boundaries separate the SoI, enabling systems, interoperating systems, and interfacing systems, supporting the SE practitioner in properly accounting for all the necessary elements which comprise the whole system context. Part of the system concept are the system's modes and states which are fundamental system behavior characteristics important to SE. Systems can be hierarchical in their structural organization, or they can be complex where hierarchy is not always present. The system concepts encompass all types of systems structures and support the SE practitioner with a framework in which to engineer a system.

1.3.1 System Boundary and the System of Interest (SoI)

General System Concepts An external view of a system must introduce elements that specifically do not belong to the system but do interact with the system. This collection of elements is called the *system environment or context* and can include the users (or operators) of the system. It is important to understand that the system environment or context is not limited to the operating environment, but also includes external systems that interface with or support the system at any time of the life cycle.

The internal and external views of a system give rise to the concept of a *system boundary*. In practice, the system boundary is a "line of demarcation" between the system under consideration, called the system of interest (SoI), and its greater context. It defines what belongs to the system and what does not. The system boundary is not to be confused with the subset of elements that interact with the environment.

The *functionality* of a system is typically expressed in terms of the interactions of the system with its operating environment, especially the users. When a system is considered as an integrated combination of interacting elements, the functionality of the system derives not just from the interactions of individual elements with the environmental elements but also from how these interactions are influenced by the organization (interrelations) of the system elements. This leads to the concept of *system architecture*, which ISO/IEC/IEEE 42020 (2019) defines as:

> Fundamental concepts or properties of an entity in its environment and governing principles for the realization and evolution of this entity and its related life cycle processes.

This definition speaks to both the internal and external views of the system and shares the concepts from the definitions of a system (see Section 1.1).

Scientific Terminology Related to System Concepts In general, *engineering* can be regarded as the practice of creating and sustaining systems, services, devices, machines, structures, processes, and products to improve the quality of life—getting things done effectively and efficiently. The repeatability of experiments demanded by science is critical for delivering practical engineering solutions that have commercial value. Engineering in general, and SE in particular, draw heavily from the terminology and concepts of science.

An *attribute* of a system (or system element) is an observable characteristic or property of the system (or system element). For example, among the various attributes of an aircraft is its air speed. Attributes are represented symbolically by variables. Specifically, a *variable* is a symbol or name that identifies an attribute. Every variable has a domain, which could be but is not necessarily measurable. A *measurement* is the outcome of a process in which the SoI interacts with an observation system under specified conditions. The outcome of a measurement is the assignment of a *value* to a variable. A system is in a *state* when the values assigned to its attributes remain constant or steady for a meaningful period of time (Kaposi and Myers, 2001). In SE and software engineering, the *system elements* (e.g., software objects) have *processes* (e.g., operations) in addition to attributes. These have the binary logical values of being either *idle* or *executing*. A complete description of a system state therefore requires values to be assigned to both attributes and processes. *Dynamic behavior* of a system is the time evolution of the system state. *Emergent behavior* is a behavior of the system that cannot be understood exclusively in terms of the behavior of the individual system elements. See Section 1.3.2 for further information on emergent behavior and Section 1.3.6 for more information on states and modes.

The key concept used for problem solving is the *black box/white box* (also known as *opaque box/transparent box*) system representation. The *black box (opaque box)* representation is based on an external view of the system (attributes). The *white box (transparent box)* representation is based on an internal view of the system (attributes and structure of the elements). Both representations are useful to the SE practitioner and there must be an understanding of the relationship between the two. A system, then, is represented by the external attributes of the system, its internal attributes and structure, and the interrelationships between these that are governed by the laws of science.

1.3.2 Emergence

Emergence describes the phenomenon that whole entities exhibit properties which are meaningful only when attributed to the whole, not to its elements. Every model of human activity system exhibits properties as a whole entity that derive from its element activities and their structure, but cannot be reduced to them (Checkland, 1999). Emergence is a fundamental property of all systems (Sillitto and Dori, 2017). According to Rousseau et al. (2018), emergence derives from the systems science concept of "properties the system has but the elements by themselves do not."

System elements interact between themselves and can create desirable or undesirable phenomena called *emergent properties* such as inhibition, interference, resonance, or reinforcement of any property. Emergent properties can also result from the interaction between the system and its environment. Many engineering disciplines include emergence as a property. For example, system safety (Leveson, 1995) and resilience (Rasoulkahni, 2018) are examples of emergent properties of engineered systems (see Sections 3.1.11 and 3.1.9, respectively).

Definition of the architecture of the system includes an analysis of interactions between system elements in order to reinforce desirable and prevent undesirable emergent properties. According to Rousseau et al. (2019), the systemic virtue of emergent properties are used during systems architecture and design definition to highlight necessary derived functions and internal physical or environmental constraints (see Sections 2.3.5.4 and 2.3.5.5, respectively). Corresponding derived requirements should be added to system requirements baseline when they impact the SoI.

Calvo-Amodio and Rousseau (2019) explain how emergence applies to systems in which complexity is dominant. Complexity dominance, they say, encourages us to consider the significance of the difference between kinds of