

JUST TELL ME WHAT I NEED TO KNOW... SYSTEM ENGINEERING

DESIGN IT. INTEGRATE IT. VERIFY IT. DELIVER IT. MAKE IT WORK.



DEFINE

Understand needs, stakeholders, and system context.



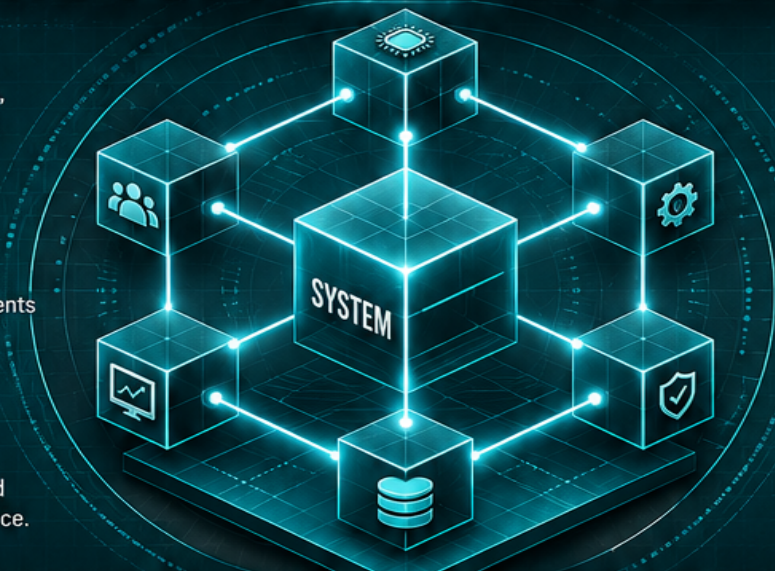
ARCHITECT

Design solutions that meet requirements and constraints.



MODEL

Use models and tools to analyze and optimize performance.



INTEGRATE

Bring components together into a cohesive system.



VERIFY

Ensure the system meets requirements and performs as intended.



EVOLVE

Monitor, learn, and improve across the system lifecycle.

CORE SYSTEM ENGINEERING PRINCIPLES



REQUIREMENTS
DRIVEN



SYSTEMS
THINKING



INTERDISCIPLINARY
COLLABORATION



RISK & TRADE-OFF
MANAGEMENT



LIFECYCLE
FOCUS



DEFENSE



AEROSPACE



IT & SOFTWARE



INDUSTRIAL



HEALTHCARE



TRANSPORTATION

PRACTICAL KNOWLEDGE. REAL-WORLD RESULTS.



BELSHAW CONSULTING

Just Tell Me What I Need To Know

Systems Engineering

A Security Architect's Practical Guide

Belshaw Consulting

Copyright Notice

This book is an independent educational work. Any company names, product names, trademarks, service marks, technologies, standards, frameworks, methodologies, or registered trademarks mentioned in this publication are the property of their respective owners. Their inclusion is for identification, educational, and illustrative purposes only and does not imply any affiliation with, endorsement by, or sponsorship from the respective owners.

Technology, products, standards, regulations, certifications, security guidance, and industry best practices evolve over time. Readers should consult current vendor documentation, official standards publications, applicable regulations, and their organization's policies and procedures before implementing any recommendations or making technical, security, or business decisions.

Copyright © 2026 Belshaw Consulting

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, transmitted, or distributed in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher, except for brief quotations used in reviews or scholarly works as permitted by applicable copyright law.

Table of Contents

Introduction	5
Part I — Systems Thinking.....	6
Chapter 1 - What Is Systems Engineering?	6
Chapter 2 - Systems Engineering Standards	15
Part II — Understanding the Problem.....	26
Chapter 3 - Stakeholder Needs	26
Chapter 4 - Requirements Engineering.....	35
Part III — Architecture and Design.....	45
Chapter 5 - System Architecture	45
Chapter 6 - Design and Trade Studies	55
Part IV — Systems Engineering Process.....	65
Chapter 7 - The Systems Engineering Lifecycle.....	65
Chapter 8 - Integration.....	75
Chapter 9 - Verification and Validation	84
Part V — Engineering Analysis	94
Chapter 10 - Systems Analysis.....	94
Chapter 11 - Engineering Economics	102
Chapter 12 - Reliability Engineering.....	111
Chapter 13 - Human Systems Integration	122
Chapter 14 - Logistics and Supportability	132
Chapter 15 - Safety, Security, and Resilience	142
Part VII — Systems Security Engineering (NIST SP 800-160)	154
Chapter 16 - Engineering Secure Systems	154
Chapter 17 - Cyber-Resilient Systems	Error! Bookmark not defined.
Part VIII — Digital Engineering.....	193
Chapter 18 - Model-Based Systems/Digital Engineering	193
Chapter 19 - Systems Engineering Management.....	204
Part IX — Modern Systems Engineering	215

Chapter 20 - Systems Engineering in the Cloud Era.....	215
Appendices.....	227
Appendix A - Systems Engineering Review Checklist.....	227
Appendix B - Requirements Engineering Checklist.....	234
Appendix C - Architecture Review Checklist	241
Appendix D - Verification and Validation Checklist.....	249
Appendix E - Configuration Management Checklist	257
Appendix F - Technical Review Checklist.....	264
Appendix G - Systems Engineering Management Plan Template	273
Appendix H - Key NIST Publications for Systems Engineers.....	282
Appendix I - Key INCOSE References for Systems Engineers	287
Appendix J - Glossary of Systems Engineering Terms	293
Appendix K - Acronyms	301
Appendix L - Recommended Reading	306

Introduction

Modern life depends on complex systems. Modern systems are everywhere.

Every time you unlock a phone with your face, board an aircraft, transfer money through a bank, drive a modern vehicle, access a cloud application, or interact with an artificial intelligence system, you are relying on technologies that must work together safely, reliably, securely, and predictably.

Making that happen is the purpose of systems engineering.

Systems engineering is the discipline of bringing together people, processes, technology, hardware, software, data, security, operations, logistics, and management to create solutions to complex problems. While individual engineering disciplines may concentrate on particular components, systems engineering focuses on the whole system - how the pieces interact, whether the solution satisfies its intended mission, and whether it can continue doing so throughout its lifecycle.

A systems engineer asks questions such as:

- What problem are we actually trying to solve?
- What do stakeholders need the system to accomplish?
- How should the system be structured?
- How do the components interact?
- What happens when something fails?
- Can the system be built, integrated, tested, operated, and maintained?
- Is it secure and resilient?
- What are the cost, schedule, performance, and technical tradeoffs?
- Will the system continue to meet its mission as requirements and technologies change?

These questions are increasingly important because modern systems are more interconnected, software-dependent, data-driven, and complex than ever before.

Whether the system is a spacecraft, defense platform, autonomous vehicle, enterprise network, cloud environment, critical infrastructure platform, or AI-enabled application, the fundamental objective remains the same:

Engineer the whole system—not just the individual pieces.

That is what this book will teach you.

Let's get started.

Part I — Systems Thinking

Chapter 1- What Is Systems Engineering?

"Systems engineering is an interdisciplinary approach and means to enable the realization of successful systems."

— INCOSE Systems Engineering Handbook

Chapter Objectives

After completing this chapter, you should be able to:

- Define what a system is.
- Explain the characteristics of complex systems.
- Distinguish between open and closed systems.
- Describe a System of Systems (SoS).
- Understand emergent behavior.
- Identify stakeholders.
- Define system boundaries.
- Explain the system lifecycle.
- Recognize why systems fail.
- Understand the role of the systems engineer.

Introduction

Every modern product is part of a larger system.

A smartphone isn't just hardware—it's an ecosystem of software, cloud services, cellular networks, applications, cybersecurity, artificial intelligence, manufacturing, logistics, and customer support.

An aircraft isn't simply an airplane. It is an integrated collection of propulsion systems, avionics, communications, navigation, cybersecurity, maintenance systems, supply chains, trained personnel, airports, satellites, and regulatory agencies.

The same is true for autonomous vehicles, hospitals, manufacturing plants, power grids, cloud computing environments, financial systems, and military weapon systems.

Each consists of hundreds—or even thousands—of interconnected components working together to achieve a common objective.

Designing these systems requires more than simply building individual parts. Someone must ensure that every component functions together as a cohesive whole.

DESIGN BETTER SYSTEMS. REDUCE RISK. DELIVER RESULTS.

Today's systems are more connected, software-driven, cloud-enabled, and security-conscious than ever before. Whether you're developing aircraft, medical devices, enterprise software, defense systems, autonomous vehicles, or critical infrastructure, systems engineering provides the framework for turning complex ideas into successful solutions.

The challenge? Most systems engineering books are either too academic, too theoretical, or too large to serve as a practical reference.

This book is different.

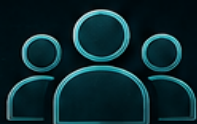
Just Tell Me What I Need To Know... **Systems Engineering** distills the essential principles, processes, and best practices into a concise, easy-to-read guide that focuses on what professionals actually need to know. It combines proven engineering fundamentals with today's modern practices, including Model-Based Systems Engineering (MBSE), Digital Engineering, DevSecOps, cloud-native architectures, cybersecurity, artificial intelligence, and cyber resilience.



INSIDE YOU'LL LEARN HOW TO:

- ✓ Understand the complete systems engineering lifecycle
- ✓ Capture and manage stakeholder needs and requirements
- ✓ Develop effective system architectures and perform trade studies
- ✓ Integrate, verify, validate, and deploy complex systems
- ✓ Manage technical risk, configuration, and engineering reviews
- ✓ Apply reliability, safety, security, and resilience engineering
- ✓ Use Model-Based Systems Engineering (MBSE) and Digital Engineering
- ✓ Engineer modern cloud, AI-enabled, and cyber-resilient systems
- ✓ Lead engineering teams and manage systems engineering programs

THE SYSTEMS ENGINEERING LIFECYCLE



Whether you're a student entering the field, an engineer expanding your skills, a project manager working with technical teams, or an experienced systems engineer looking for a practical refresher, **this book provides the knowledge you need—without unnecessary complexity.**

KEY BENEFITS YOU'LL GAIN



CLARITY

Understand core principles without unnecessary theory.



CONFIDENCE

Apply proven methods to real-world challenges.



INTEGRATION

Connect people, processes, and technology.



RISK REDUCTION

Manage technical risk and improve system outcomes.



BETTER DECISIONS

Use trade-offs and data to make informed choices.



DELIVER RESULTS

Build systems that meet needs, on time and on budget.



**LESS THEORY.
MORE PRACTICAL GUIDANCE.
JUST TELL ME WHAT I NEED TO KNOW.**



BELSHAW CONSULTING