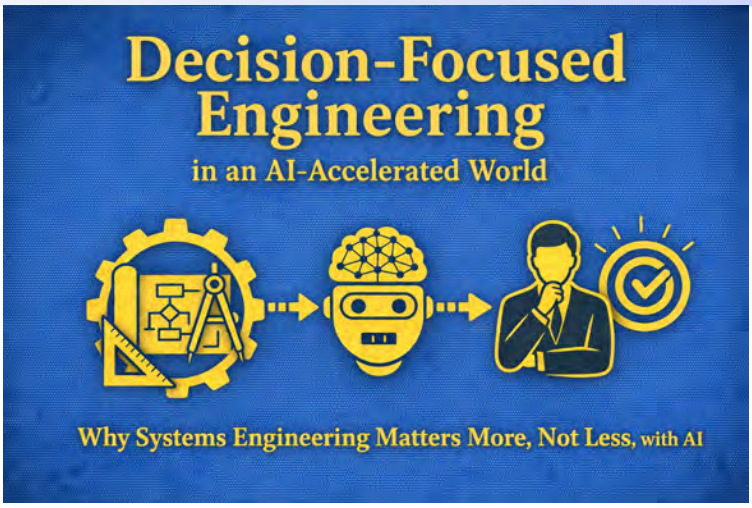


Dr. Jerry Jon Sellers

Teaching Science & Technology Inc.

Implications for Systems Engineers, Chief Engineers, and Program Teams

Presenter—Dr. Jerry Sellers, Ph.D., University of Surrey, UK (jsellers@tsti.net)

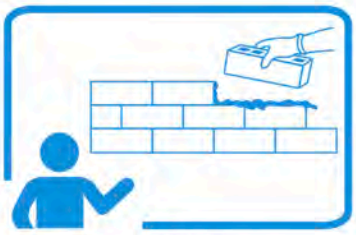
- Partner and President, Teaching Science and Technology Inc. (TSTI) with over 40 years of aerospace engineering experience, including Space Shuttle Mission Controller, NASA/JSC, and Chief of Astronautics for the European Office of Aerospace Research and Development, London, UK
- Former Director of the Space Systems Research Center, USAF Academy. Responsible for designing, building, testing and operating small, scientific satellites.
- Author of the text *Understanding Space: An Introduction to Astronautics*, Editor and contributing author of the text *Applied Space Systems Engineering*
- Adjunct Professor, Stevens Institute of Technology.
- Certified Agile/Scrum Master, Certified Scaled Agile Framework (SAFe) Program Consultant (SPC6)
- Associate Fellow AIAA, Elected Member - International Academy of Astronautics

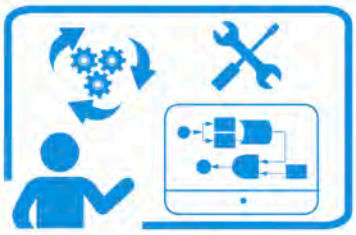
About TSTI

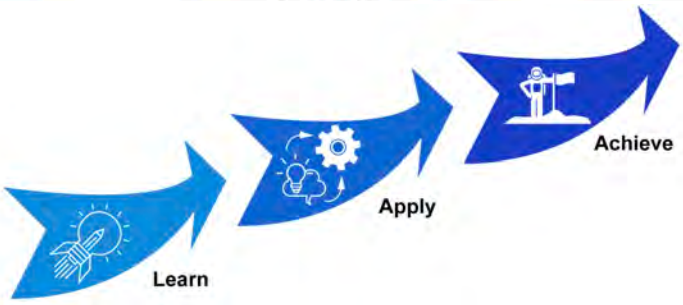
- Teaching Science & Technology, Inc. (TSTI) is your space systems engineering capability partner.
- For more than 30 years we've helped organizations build disciplined engineering maturity—from early mission architecture through execution—by embedding practical systems thinking into their people, processes, and programs.
- Our work spans commercial space companies, defense organizations, and national space agencies wanting to build capacity by developing their technical workforce.
- Through integrated courses, workshops, consulting engagements, and applied frameworks such as the Decision Framework and Essential Lifecycle Model, we enable teams to make better architectural decisions, align proposal promises with execution realities, and implement digital engineering in ways that actually improve outcomes.
- TSTI is a globally engaged small business, bringing decades of applied aerospace experience to organizations that are serious about building effective space capability.
- Achievements
  - Presented thousands of course to tens of thousands of engineers in dozens of countries (MELCO, JAXA, NOAA, NASA, NRO, USSF, ESA, UAESA,, GGPEN, CSA, industry)
  - Alumni hold top technical management positions in government and industry

TSTI Courses

Our courses span the range from basic foundational training, to tools and techniques, to advanced applications





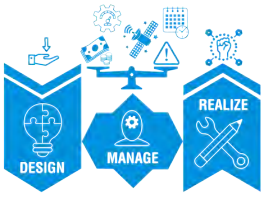


TSTI Courses and Workshops—Live, Virtual or Online*

Details on each of these and other courses can be found at www.tsti.net

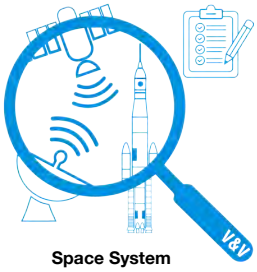
Understanding Space

Designing Space Missions and Systems

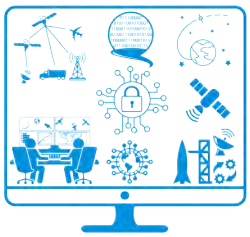


Applied Space Systems Engineering

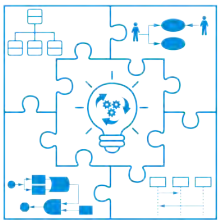
Remote Sensing Systems—Fundamentals and Applications



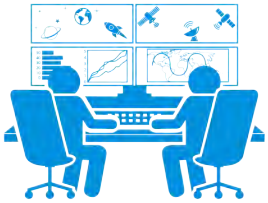
Space System Verification & Validation



Space Domain Cybersecurity



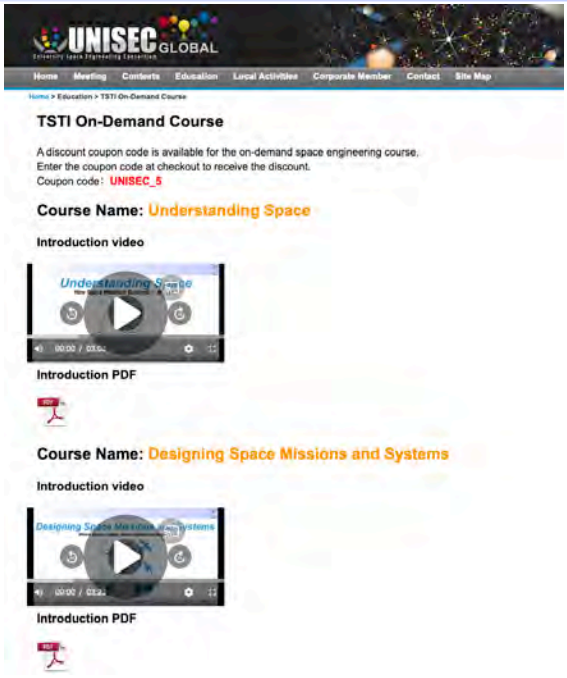
Model-based Systems Engineering (MBSE)/ Digital Engineering



Space Mission Operations

*Not all courses are currently available online

Special Offer through UNSEC



The Moment We're In

- Artificial intelligence is now part of everyday engineering work.
- Across the space and aerospace sector, teams are already using AI to:
  - generate concepts and architectures
  - accelerate trade studies and analysis
  - draft requirements and documentation
  - support proposal development
  - explore design alternatives

- The pace of engineering work is increasing.
- The number of options available to teams is increasing.
- But the fundamental responsibility for program outcomes has not changed.
  - There's no escaping the "systems engineering universe" that must balance cost, schedule, performance and risk

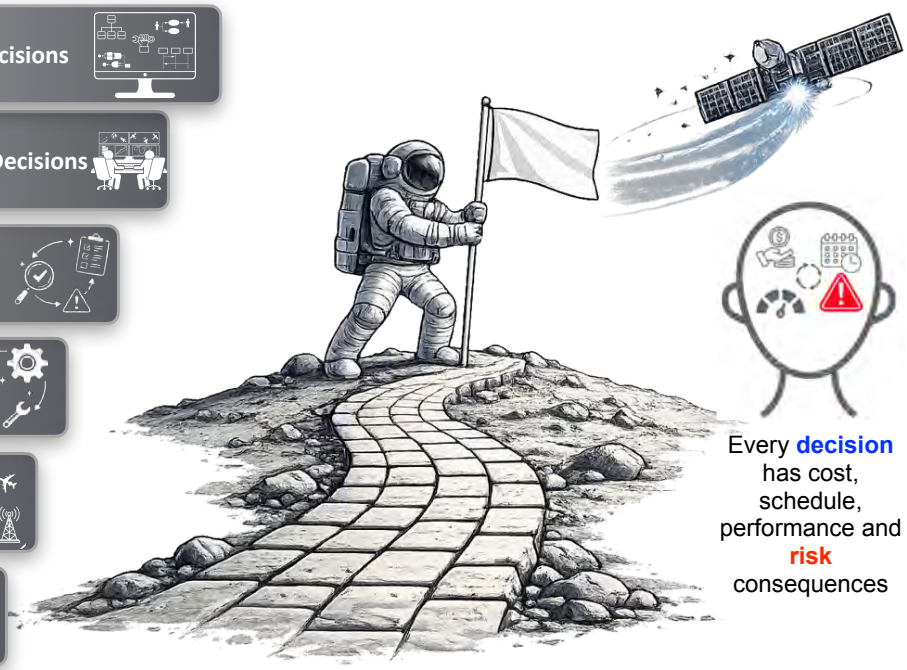
The Common Narrative

- Much of the current discussion on AI focuses what it will automate:
  - "AI will design systems."
  - "AI will replace engineers."
  - "AI will automate systems engineering."

- These statements miss the central issue.
- AI is accelerating the *creation* of engineering artifacts.
- *It is not assuming responsibility for program decisions.*
- Engineering programs still succeed or fail based on the quality of the decisions that teams make and commit to.

The Road to Project Success is Paved with Decisions

- Information & Insight Decisions
- Operational & Sustainability Decisions
- Evidence & Confidence Decisions
- Execution & Rigor Decisions
- Architecture & Trade-off Decisions
- Feasibility & Context Decisions



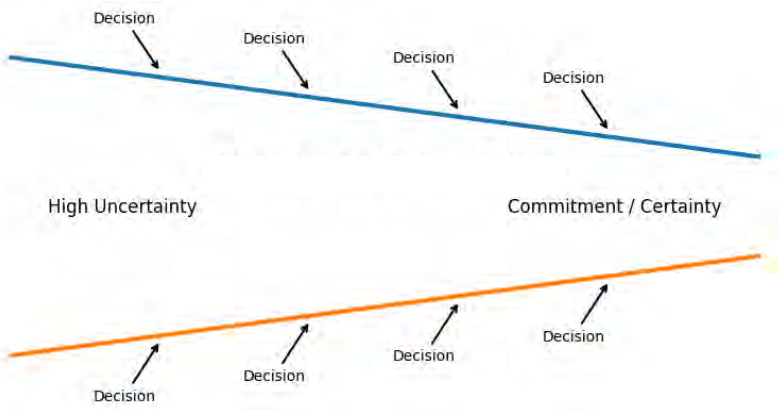
Every **decision** has cost, schedule, performance and **risk** consequences

Core Premise

- ✦Projects as an enterprise do not exist to produce artifacts.
- ✦*Projects exist to enable informed decisions that commit time, money and other resources with the promise of a delivered capability, and accept risks in various forms in exchange*
- ✦Requirements, architectures, models, and analyses are not ends in themselves.
- ✦They are developed to support understanding, alignment, and decision-making.
- ✦Engineering progress is measured not by the volume of artifacts produced, but by the quality and timing of the decisions that move a program forward.
- ✦This becomes more important and true in an AI-accelerated environment.

Projects as Structured Decision Enterprises

- ✦Every project moves from uncertainty toward commitment driven by a series of decisions.
- ✦At each stage, teams must:
  - define objectives and intent
  - establish constraints
  - generate and evaluate options
  - analyze performance and risk
  - align stakeholders
  - commit to a course of action



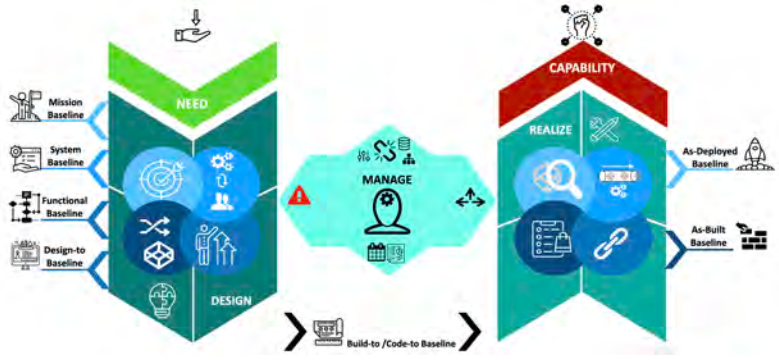
Options Narrow as Decisions are Made

Project Decision Domains

- ✦Feasibility and Context—*Is this mission possible, and under what constraints?*
- ✦Architecture and Trade-off Decisions—*What system should we build, and why?*
- ✦Execution and Rigor Decisions—*What Processes and How much discipline is required to deliver this system?*
- ✦Evidence and Consequence Decisions—*What proof do we need before we fly?*
- ✦Operational and Sustainability Decisions—*Can this mission actually be operated day after day within real constraints?*
- ✦Information and Insight Decisions—*How do we structure information to support good decisions?"*

Classic Systems Engineering Framework

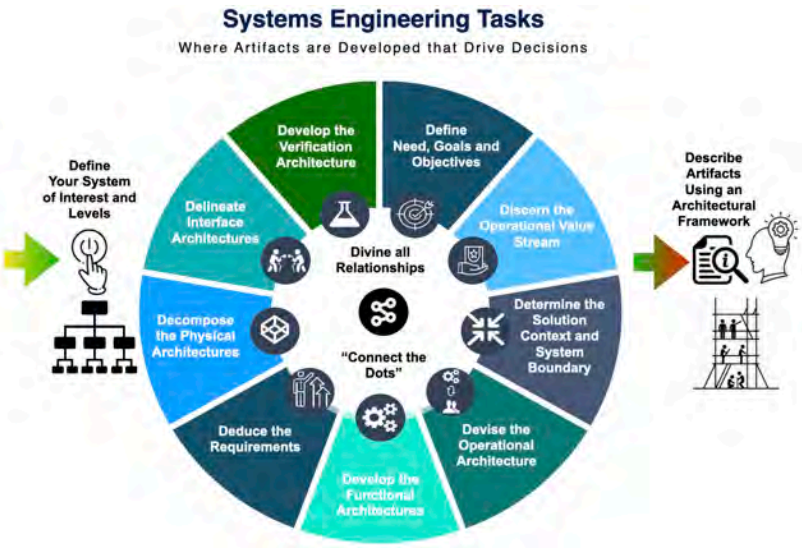
- Systems Engineering can be modeled as a Framework of highly inter-related activities with the goal of creating real products delivering real capabilities (not just churning out documents)
  - Think Framework..not simply Process
  - "Left Side" activities aimed at Design
  - "Right Side" activities aimed at Realization
  - Both supported by a variety of SE management activities
  - All help us increase our knowledge of the system from one technical baseline to the next



Systems engineering is first and foremost about getting the right design—and then about maintaining and enhancing its technical integrity, as well as managing complexity with good processes. Neither the world's greatest design, poorly implemented—nor a poor design, brilliantly implemented—is worth having.

Systems Engineering as a Decision "Engine"

- Systems engineering provides the structure that allows these decisions to be made coherently.
- Engineering artifacts exist to support decisions.
- Decisions shape the system that is ultimately built.



Systems Engineering Tasks 🔁 Decision Domains

- ✦Systems engineering organizes information so teams can make coherent decisions.
- ✦Systems engineering produces and integrates:
  - requirements
  - functional and physical architectures
  - trade studies
  - performance analyses
  - verification and validation plans
- ✦These artifacts help teams understand options, evaluate tradeoffs, and manage risk.
- ✦They provide evidence to the people who must ultimately commit to technical and programmatic choices.
- ✦**Systems engineering is the engine that drives decision support across the lifecycle.**

The Decision Framework organizes what trade-off decisions must be made; Systems Engineering Tasks produce the artifacts that enable those decisions with structure, evidence, and rigor.

Where AI Enters the Engineering Process

- ✦AI is increasingly integrated into the creation of engineering artifacts.
- ✦Teams are using AI to accelerate:
  - concept and architecture generation
  - trade study exploration
  - analysis setup and iteration
  - requirements and documentation drafting
  - proposal development
  - product development
  - sustainment

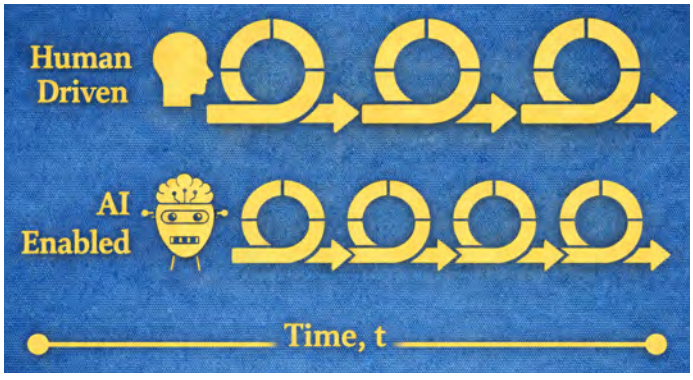
- ✦Currently and near-term, AI operates primarily within the artifact-generation and analysis loop.
  - It increases speed, breadth of options, and iteration rate across early phases of programs.

What AI Does NOT Do

- ✦AI can generate options in the form of various artifacts quickly.
- ✦It does not assume responsibility for program outcomes. AI does not (currently):
  - define mission intent
  - establish program constraints
  - set risk posture
  - align with stakeholders
  - make commitments on behalf of the team
- ✦AI produces possibilities.
- ✦Humans define parameters and make decisions.

Early-Phase Impact

- ✦The greatest impact of AI is occurring in early program phases:
  - proposal development
  - requirements development and analysis
  - mission concept exploration
  - architecture definition
  - pre-PDR trade studies
- ✦AI enables:
  - faster concept iteration (more design loops)
  - broader exploration of alternatives
  - shorter timelines for analysis
- ✦This increases the need for clarity, alignment, and disciplined decision-making early in the lifecycle.



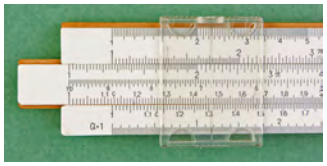
Explosion of Options

- Historically, teams explored a limited set of design options due to time and resource constraints.
- With AI-assisted tools:
  - more options can be generated
  - more trades can be explored
  - iteration cycles are faster

- The bottleneck is no longer option generation.
- The bottleneck is decision quality and alignment.
- More options increase the importance of structured evaluation and commitment.

The Evolving Chief Engineer

- The chief engineer's role becomes even more central in an AI-enhanced environment.
- Responsibilities increasingly include:
  - framing the decision space
  - defining constraints and assumptions
  - *validating AI-generated outputs*
  - maintaining architectural coherence
  - guiding trade decisions and convergence
- The chief engineer integrates human expertise and AI-enabled analysis to support sound program decisions.



To use a slide rule, engineers needed to know the answer within an order of magnitude!

The Evolving Program Manager

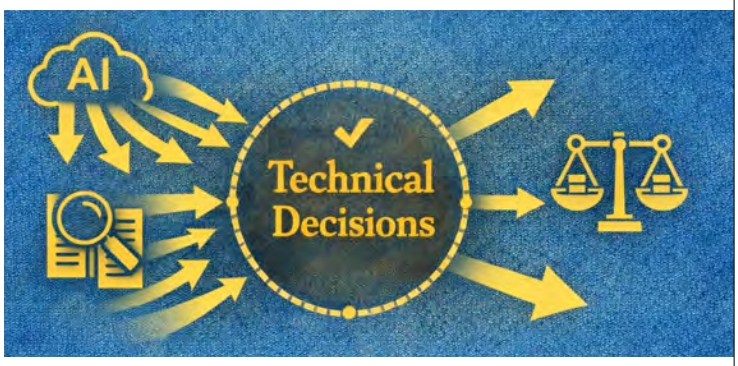
- ✦Program managers remain responsible for commitment, alignment, and risk posture.
- ✦AI accelerates technical analysis and artifact creation. It does not replace program accountability.
- ✦Program manager responsibilities include:
  - aligning stakeholders around decisions
  - managing program risk and commitments
  - ensuring decisions are understood and owned
  - balancing cost, schedule, and performance
- ✦AI accelerates analysis. It does not assume accountability for outcomes.

Risks of Misapplied AI

- ✦AI can accelerate engineering work, but misuse introduces real risks.
- ✦Common risks include:
  - shallow or incomplete trade studies
  - overconfidence in plausible outputs
  - architecture drift across teams
  - requirements instability and churn
  - analysis without clear ownership
- ✦Plausible outputs are not the same as validated decisions.
- ✦Speed can mask misunderstanding if teams do not maintain discipline.

Practical Guidance for Teams

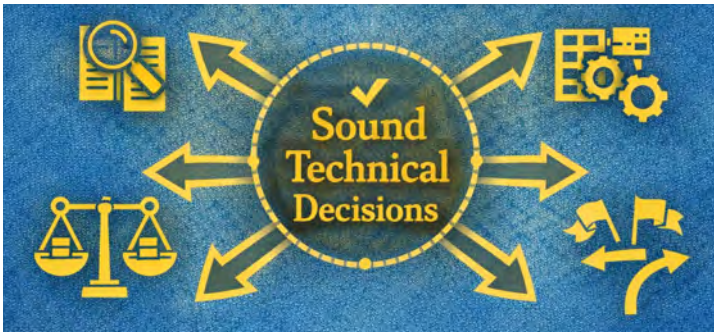
- ✦AI can be a powerful contributor when used deliberately.
- ✦Use AI to:
  - generate options and alternatives
  - explore trade spaces
  - draft and iterate artifacts
- ✦But teams should:
  - define constraints and intent first
  - validate outputs rigorously
  - maintain architectural baselines
  - document decisions and assumptions
  - avoid premature convergence



- ✦AI can accelerate preparation for decisions, but doesn't necessarily speed up the decisions themselves.

Implications for Training and Workforce

- ✦As AI accelerates artifact generation, the most valuable skills shift.
- ✦Training emphasis should move from:
  - producing artifacts
- ✦toward:
  - framing problems clearly
  - evaluating options and trades
  - validating results
  - integrating across domains
  - making and defending decisions

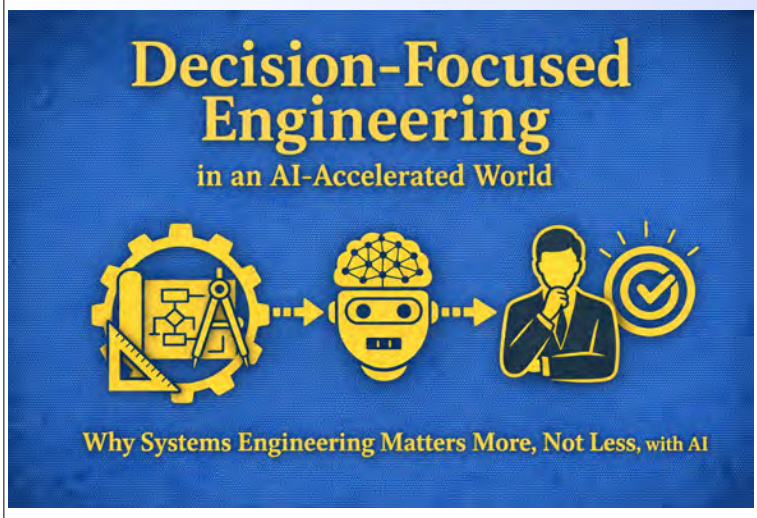


- ✦The defining engineering skill becomes the ability to make sound, well-informed technical decisions.

Closing: What Changes and What Doesn't

- AI is changing the pace of engineering work.
- It is increasing the number of options available to teams.
- It is not changing who is responsible for decisions.
- Programs will still succeed or fail based on:
  - clarity of intent
  - quality of decisions
  - alignment of stakeholders
  - ownership of commitments
- *In an AI-accelerated world, systems engineering matters more—not less—because the ultimate decisions (and consequences thereof) still belong to humans.*

Questions? Discussion?



Thank you for your time and attention

Dr. Jerry Jon Sellers

Teaching Science & Technology Inc.

Implications for Systems Engineers, Chief Engineers, and Program Teams