INCOSE Michigan Chapter
Monthly Newsletter
May 2026

In This Issue

- President's Message
- Chapter News
- Upcoming Events Calendar
- Upcoming Chapter Events
- Partner Organization Events
- Other INCOSE Chapter Events
- Other Events
- Historical Corner

President's Message

April was another bustling month for INCOSE Michigan. I attended our Engineering Expedition at the Packard Proving Grounds on April 12th. The presentation on The Rise and Fall of the DeLorean Motor Company was interesting; it brought to mind the old adage, “One miracle per program.” The design, factory construction, workforce training, economic and quality challenges facing the DeLorean Motor Company were individually challenging; collectively, they were insurmountable. The visit also brought to mind memories of my parents’ Packards and prompted one of this month’s historical articles. I have included a few DeLorean photos to give a sense of this iconic vehicle.

Our sold-out SysML 2.0 bootcamp was a great experience for all who attended. We appreciate the hospitality shown to us by Dassault Systèmes and are grateful that they provided the instructors and venue for this event. The attendees came away from this three-day experience with an understanding of the power of SysML 2.0 and the changes it will bring to the practice of system engineering.

Our Michigan Technological University student section presentation is detailed below. NASA recognized their efforts with the APEX award and a challenge coin…well done!

Jeremy Ross will be our May in-person speaker; his talk is entitled *Addressing Siloed Risk Analysis: Developing a Holistic Safety and Reliability Ecosystem using MBSE and the Risk Assessment and Modeling Language (RAAML).* Details and registration link for this May 12th event are below.

We are launching an Ambassador Program to help bring information and excitement about our activities to local organizations; see below for details.

Finally, June will be another SysML 2.0 month! June 5th is the System-as-Code Fest, hosted by System Strategy, Inc. in partnership with Sensmetry. June 6th is our SysML 2.0 vendor showcase and we wrap up the month with a June 24th presentation by SysIDE.

I hope you take advantage of some of this cutting-edge programming!

Michael J. Vinarcik, ESEP-Acq, P.E, FESD
President

A DeLorean with optional flux capacitor upgrade

DeLorean (flat hood version)

Original DMC engine

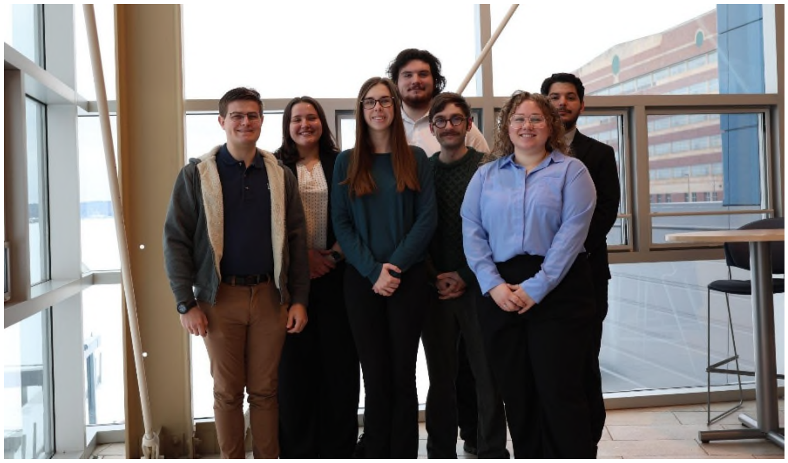
Corvette Engine fitted into a DeLorean

The sold-out SysML 2.0 Bootcamp at Dassault Systèmes Royal Oak

Chapter News

NASA Exploration Division Presents APEX Award to MTU–INCOSE-NASA EX2 Capstone Collaboration Team

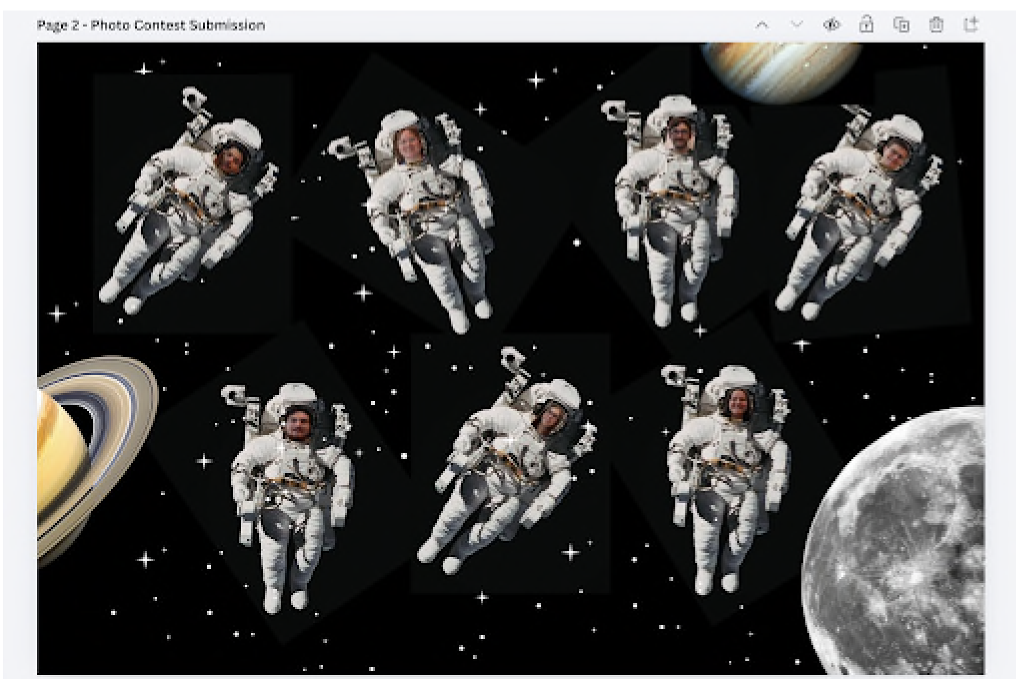
This year's MTU–INCOSE–NASA Capstone Collaboration marked a standout achievement for the students and mentors who took on this complex systems-engineering challenge. The project's customer was NASA's Spacecraft Performance and Concept Engineering Branch within the Exploration Division at Johnson Space Center. Each year, the project requires students to work directly with NASA stakeholders each week while applying the full breadth of their systems engineering training — and for four consecutive years, the teams have delivered extraordinary results.



From left to right: Nicholas Wyatt, Kathryn Bugbee, Katie Walters, Connor Jones, Tim Peters, Ella Santi, Brian Marku

The team was given an open problem statement. **To provide value to NASA from when the Astronaut leaves the atmosphere to when the Astronaut returns**. The project titled, **Conceptual Evaluation of CryoMag Sustainable Operational Usage with NASA** was defined, planned, managed, executed, verified, and delivered in a single semester. Each team member performed with technical excellence, inspiration towards a common goal, and with a true understanding of how to bring together each other's interests and strengths to achieve the mission of Space Exploration.

This year, the team's exceptional performance did not go unnoticed. The team received two awards at the Michigan Technological University's Senior Design Expo, with a 2nd place overall and a 2nd place award for the team photo In a special recognition ceremony, the **NASA Exploration Division** presented the entire Capstone team — along with the **INCOSE Academic Chair and Faculty advisors** — with the **APEX Award for Awesome Performance in the Exploration Division**. This award is reserved for teams that demonstrate outstanding technical rigor, innovation, and collaboration in support of NASA's mission.



Capstone Team Senior Design 2nd Place Award Winning Photo

Adding to the significance of the moment, NASA representatives presented each student and advisor with **official NASA challenge coins**, a long-standing tradition symbolizing excellence, respect, and meaningful contribution to the Exploration Division at NASA. For many team members, receiving a challenge coin directly from NASA leaders was a powerful affirmation of their hard work and professionalism.

The Capstone Collaboration is designed to mirror the real-world environment of systems engineering: evolving requirements, interdisciplinary coordination, and the need for clear communication across technical boundaries. With guidance from NASA, MTU faculty experts, and the INCOSE Academic Advisor, the team navigated these challenges with maturity and creativity, producing work that exceeded expectations and strengthened the partnership between NASA and the academic program.

This recognition not only celebrates the students' achievements but also highlights the value of the ongoing partnership among NASA, MTU, and INCOSE in preparing the next generation of systems engineers. The APEX Award stands as a testament to the team's commitment, the advisors' leadership, and the impact of mission-driven, collaborative education.

Congratulations to the entire team, Katie Walters, Brian Marku, Connor Jones, Nicholas Wyatt, Ella Santi, Tim Peters, Kathryn Bugbee, the INCOSE Academic Chair Robin Mikola, the MTU faculty advisor James Bittner, and Jon Sticklen, the INCOSE MTU Student Division Faculty Advisor and cofounder of the project with Robin Mikola, for an extraordinary accomplishment.

INCOSE Michigan Ambassador Program

The Ambassador Program connects the INCOSE Michigan Chapter with key organizations across Michigan—employers, universities, and professional societies where systems engineers work and learn. Karl Selewski, Vice-President, is coordinating their activities; please contact him at karl.selewski@incose.net if you would like to volunteer as an Ambassador. It is a great way to get involved (and earn CEUs for recertification).

Thanks to our initial 2026 Ambassadors:

- Radu Babiceanu — Western Michigan University
- Bill Berklich — Ground Vehicle Systems Center
- Prajakta Chavan — Borg-Warner/Oakland University
- Carl Jolma — Michigan Technological University
- Ravindra Kale — General Motors
- Robin Mikola — System Strategy, Inc.
- Binod Pant — Dauch Corporation
- Jeremy Ross — Ford
- Anand Wanjari — Cummins/KPIT
- Ozzi Weli — Aptiv

Upcoming Events Calendar

Date	Event	Organization
May 12	Addressing Siloed Risk Analysis: Developing a Holistic Safety and Reliability Ecosystem using MBSE and the Risk Assessment and Modeling Language (RAAML)	INCOSE Michigan Chapter
May 12	Solving Programmatic Risks with Ken Cureton	INCOSE Los Angeles Chapter
May 22	The Strategic Mindset – For Your Organization, Your Career and Your Life	PMI Great Lakes Chapter
May 27	Mascots in Motion	Stahls Automotive
June 5	System-as-Code Fest	Sensmetry/System Strategy, Inc.
June 6	SysML 2.0 Vendor Day	INCOSE Michigan Chapter
June 13	Packards & Pours	Packard Proving Grounds
June 24	SysIDE SysML v2.0	INCOSE Michigan Chapter

Upcoming Chapter Events

Addressing Siloed Risk Analysis: Developing a Holistic Safety and Reliability Ecosystem using MBSE and the Risk Assessment and Modeling Language (RAAML)

Date: May 12, 6:00 PM - 8:00 PM (Dinner provided)

Location: The Engineering Society of Detroit, 20700 Civic Center Drive, Suite 450, Southfield, MI 48076

Model-Based Systems Engineering (MBSE) replaces siloed, document-centric processes with integrated digital workflows, establishing a shared digital model as the authoritative design description rather than a collection of disconnected documents. While modern safety and reliability methods increasingly leverage these digital models as a baseline, they often lead to duplicative risk assessments due to unique representations of risk across analysis types such as Failure Mode and Effects Analysis (FMEA), Functional Safety (FuSa), and System Theoretic Process Analysis (STPA). This fragmentation obscures critical interdependencies between system functions and risk evaluations, potentially masking emergent failure modes and creating significant manual overhead to synchronize data as the architecture and risk assessments evolve.

This presentation demonstrates how the Risk Assessment and Modeling Language (RAAML) — an OMG-standardized modeling language designed to formally represent safety and reliability analyses within a system model — can be leveraged to consolidate these standalone workstreams into a holistic safety ecosystem, starting with Failure Mode and Effects Analysis. The presentation will explore how RAAML-compliant model-based failure mode avoidance can be integrated with descriptive system architecture models to enable inter-model read-across and reuse of both risk and architecture elements at scale. It will also outline how the comprehensive nature of RAAML enables the integration of additional risk analysis types into this holistic ecosystem as MBSE tooling and industry adoption continue to mature. This transformative approach integrates separate safety workstreams into a unified, verifiable representation of system risk that is intrinsically linked to the design intent.

Register at: https://lp.constantcontactpages.com/ev/reg/xtdtru4

About the Presenter

Jeremy Ross is an engineering supervisor at Ford Motor Company, leading development of the company's model-based systems engineering (MBSE) and failure mode avoidance tools and methods. He holds degrees in Mechanical Engineering (BSE, MSE) from the University of Michigan and in Product Development (MS) with a Systems Engineering specialization from the University of Detroit Mercy. Jeremy is a Certified Systems Engineering Professional (CSEP) and an adjunct instructor at the University of Detroit Mercy, where he lectures on MBSE and SysML. Jeremy is a frequent contributor to conferences and technical publications in the systems engineering domain, with honors including a best paper at the 2022 INCOSE International Symposium, 2024 System Modeler of the Year (Zuken/Vitech Integrate24), and 2026 Engineering Society of Detroit Young Engineer of the Year.

Engineering Expedition: Fuel for Thought: Mascots in Motion

Date: Wednesday, May 27, 2026
Time: 5:00 – 8:00 p.m.
Location: Packard Proving Grounds Historic Site, 49965 Van Dyke Avenue, Shelby Township MI 48317

Steve Purdy is the author of *Mascots in Motion*, a meticulously researched book that tells the story of the iconic hood ornaments that once defined luxury automobiles, classic cars, and brand identity. From elegant goddesses to powerful symbols of speed, innovation, and prestige, these automotive mascots reflected the design trends, cultural values, and craftsmanship of their era.

This engaging talk blends automotive history, car design, industrial art, and storytelling, offering insight into how hood ornaments became rolling works of art. Perfect for classic car enthusiasts, collectors, designers, historians, and automotive fans, this Fuel for Thought event celebrates the artistry and legacy behind some of the most recognizable symbols in automotive history.

Cost: Free

Register at https://www.stahlsauto.com/event/fuel-for-thought-mascots-in-motion-a-night-with-steve-purdy/

SysML v2 Vendor Showcase

Date: June 6, 2026

Location: The Engineering Society of Detroit, 20700 Civic Center Drive, Suite 450, Southfield, MI 48076

SysML v2 tool vendors will converge on the Engineering Society of Detroit on June 6 for a full-day showcase that goes well beyond the usual vendor demo day. Each vendor was given one of two Michigan-relevant automotive engineering challenges — an EV Battery Management System or a Power Window Controller with anti-pinch safety compliance — and has been building a complete SysML v2 model over the past two months. On event day, they will walk through their models, demonstrate their tools side-by-side, and then do something that has rarely been attempted at this scale: import each other's models and show what survives the round-trip.

Confirmed Vendors:

- Celedon Systems
- Dalus.io
- Dassault Systèmes
- Mgnite, Inc.
- Sensmetry/SysIDE
- Sysmodeler

What to Expect

Morning sessions feature 35–40 minute vendor demonstrations, each using their challenge model to walk through requirements, structure, behavior, parametrics, and tool-specific differentiators. The BMS scenario covers battery thermal management, CAN bus interfaces, and state-of-charge estimation. The Power Window Controller scenario involves FMVSS 118 anti-pinch regulatory compliance, motor force modeling, and safety-critical state machines — and is particularly well-suited for vendors wanting to demonstrate simulation or CAD integration capabilities.

The afternoon interoperability showcase is where this event stands apart. Prior to the event, vendors will have exchanged models through a shared repository and attempted cross-tool imports. During the afternoon session, they will present these results using a common scorecard: what elements transferred successfully, what required manual adjustment, and what was lost in translation. This is a collaborative exercise — the goal is to give the community an honest, structured picture of where SysML v2 interoperability stands today.

Why Attend

If you are evaluating SysML v2 tools, planning an MBSE transition, or simply want to understand where the new standard stands, this event will deliver more actionable information in one day than months of vendor webinars. You will see the same engineering problems modeled in seven competing tools, with real interoperability data — not marketing slides.

For those who attended our sold-out SysML v2 3-Day Bootcamp with Dassault Systèmes in April, this is the natural next step: apply what you learned to critically evaluate the tools available to you.

Virtual Attendance and Watch Parties

Can't make it to Southfield? A livestream will be available to registered attendees. We are also coordinating watch parties with other INCOSE chapters — if your chapter is interested in hosting one, please contact us at michael.vinarcik@incose.net.

Register: https://lp.constantcontactpages.com/ev/reg/uksxc9u

Sidequest Social: Packards & Pours

Date: Saturday, June 13, 2026
Time: 6:00 – 10:00 p.m.
Location: Packard Proving Grounds Historic Site, 49965 Van Dyke Avenue,
Shelby Township, MI 48317

Craft beer, wine, cocktails — and classic cars under the evening sky at one of Michigan's most atmospheric automotive heritage sites. Packards & Pours is the Packard Motor Car Foundation's signature summer fundraiser (formerly Packards & Pints), set among the original 1928 Tudor Revival buildings where Packard engineers once tested luxury automobiles on a 2.5-mile oval track.

Tour the historic Lodge, Repair Garage, and WWII Tank Test Center while enjoying drinks, food, and live music. All proceeds support the continued restoration of this 17-acre landmark. This is a relaxed, social evening — an ideal opportunity for chapter members and guests to mingle in a setting that practically defines Michigan automotive heritage.

Cost: $50

Register at https://packardprovinggrounds.org/event/packards-pours/

Save the Date: Sensmetry/SysIDE: SysML v2.0

Date: June 24, 2026

Location: Virtual

Registration: TBD

Partner Organization Events

PMI Great Lakes Chapter

The Strategic Mindset – For Your Organization, Your Career and Your Life

Date: Friday, May 22, 2026, 12:00 PM - 1:00 PM

Location: Virtual

Strategic planning is now embedded in PMI's Talent Triangle as a critical competency — and it's equally relevant to systems engineers managing complex programs, navigating career growth, or aligning life goals. This session explores how to build and execute a strategic plan across three dimensions: your organization, your career, and your life. Attendees will leave with the foundation of their own Professional Strategic Plan.

David Barrett is a professional speaker, author of six books, podcast host, and project management education advisor. He is the creator of ProjectWorld, ProjectTimes.com, and the online content platform ProjectBites.com, and manages virtual conferences for PMI chapters.

Cost: $20

Registration closes: Tuesday, May 19, 2026, at 12:00 PM

Register: https://pmiglc.org/calendar?eventId=44982

Other INCOSE Chapter Events

INCOSE Los Angeles Chapter: Solving Programmatic Risks with Ken Cureton

Date: Monday, May 12, 2026, 8:30–10:30 PM EDT

Format: Virtual

Major engineering systems are often heavily influenced by political processes within governments and large corporations, particularly around funding and approval. While system architects are well trained in analytical methods for cost, schedule, and performance challenges, they are often unprepared for the complexities of governmental or corporate oversight. This lecture offers system architects an overview of programmatic risks — especially political factors — to help them better understand and navigate the external forces that shape engineering decisions. A companion case study illustrates the real-world impact of political processes and explores potential risk mitigation strategies.

Speaker: Kenneth L. Cureton holds a BS in Physics from Cal State LA and an MS in Systems Architecting and Engineering from USC, where he has taught as a part-time lecturer since 1996. He retired from Boeing after 29 years and chairs the INCOSE Resilient Systems Working Group.

Register: https://lp.constantcontactpages.com/ev/reg/8545zsw

Other Events

System-as-Code-Fest 2026

Date: Thursday, June 5, 2026, 10:00 AM – 12:45 PM EDT

Format: Hybrid — in-person in Detroit, MI, or virtual via Zoom

Cost: Free

Sensmetry and Detroit-based System Strategy, Inc. are co-hosting the 2026 System-as-Code-Fest, a half-day mini-conference focused on SysML v2 and the System-as-Code paradigm. The morning session features presentations from 10 SE practitioners and thought leaders on how they are applying SysML v2 today, including in AI-enabled workflows. Following last year's event — which drew over 300 registrants and nearly 2,000 YouTube views — this year's edition brings the conference to our own backyard. On-site attendees can also join an afternoon hands-on SysML v2 training session led by Sensmetry's Kestutis Jankevicius.

Register: https://sensmetry.com/system-as-code-fest/

Greenfield Village: Salute to America

Dates: July 2-5, 2026

The iconic annual Independence Day celebration by the Detroit Symphony Orchestra and Greenfield Village, focused on America's 250th anniversary.

Registration: https://www.thehenryford.org/visit/things-to-do/calendar/salute-to-america

The Henry Ford Museum of American Innovation: Freedom Plane

Dates: July 9-26, 2026

Traveling for the first time together in history, see documents like George Washington's, Alexander Hamilton's and Aaron Burr's Oaths of Allegiance (1778); the Treaty of Paris (1783); Senate Markup of the Bill of Rights (1789); and more. The Henry Ford is one of eight institutions across the U.S. where nine original founding-era documents will be on display to the public as part of The National Archives and Records Administration traveling exhibition, Freedom Plane National Tour: Documents That Forged a Nation.

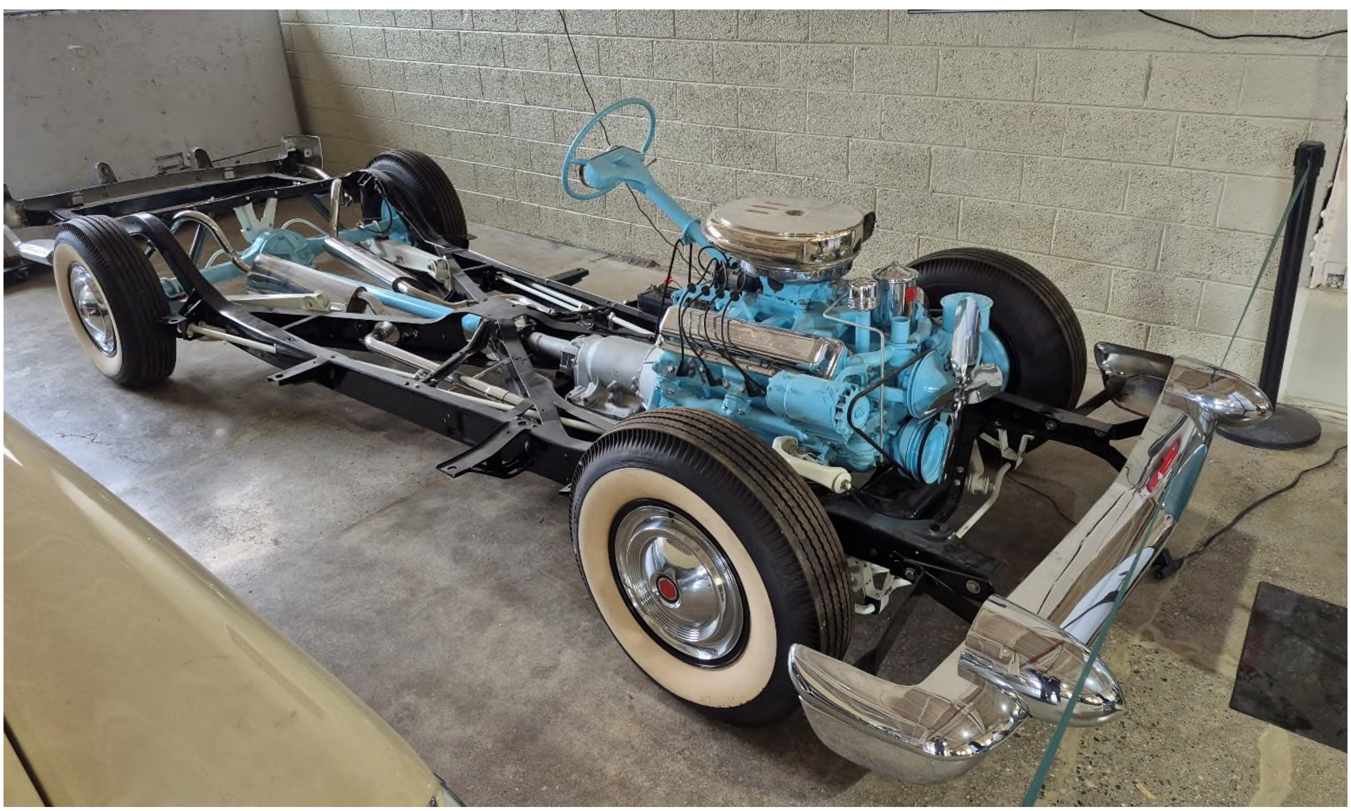
Information: https://www.thehenryford.org/visit/things-to-do/calendar/freedom-plane-national-tour

Historical Corner

Packard's Torsion-Level Ride — When Architecture Outran the Enterprise

May 2026 — 71 Years Since Introduction

In 1955, the Packard Motor Car Company introduced one of the most technically ambitious chassis systems ever offered on an American production automobile: Torsion-Level Ride. It remained in production for only two model years before Detroit-built Packard manufacturing ended in June 1956. The system's short life had little to do with its engineering merit and everything to do with the enterprise context surrounding it — a cautionary lesson as relevant to modern systems engineers as to mid-century chassis designers.



Packard factory chassis demonstrator showing the Torsion-Level suspension

The Problem: Emergent Dynamics of the Postwar American Car

By the early 1950s, American luxury cars had converged on a common formula: long wheelbases, heavy bodies, and very soft coil springs tuned for boulevard ride comfort. The soft spring rates provided a lush, luxurious ride but carried unfortunate handling traits: excessive pitch and roll, nose-dive under braking, squat under acceleration, and low-frequency “porpoising” over undulating roads. Stiffening the springs cured the handling but destroyed the ride — a classic coupled-requirements problem with no point solution.

The Architectural Response: Coupling as a Design Choice

The real innovation of Torsion-Level Ride was coupling the front and rear suspension with a pair of chassis-length torsion bars that tied the front and rear wheels together on each side. Coupled front/rear suspension was not unique — the original Citroën 2CV used a variant — but it was rare. A single long torsion bar on each side of the chassis (106 inches on Junior models, 111 inches on Seniors) acted as both the primary spring and the mechanical coupling between axles.

The consequence was a deliberate redistribution of how road inputs propagated through the system. Coupling the front and rear wheel motions significantly reduced the pitch frequency of the chassis (as well as its skew rate), permitting soft spring rates without the poor body-control effects. Rather than treating pitch as an unavoidable emergent property of independent front and rear subsystems, Packard’s engineers architected it out.

The concept originated with Hudson engineer William D. Allison in the early 1940s. Hudson’s management considered the design promising but concluded that developing it for production would cost more than the company could afford. Rather than shelve the idea, Hudson released the patent rights to Allison and granted him a leave of absence to shop the concept elsewhere. Packard licensed the patent in 1951, authorized development in 1952, tested the system on several experimental cars in 1954, and released it for production in 1955 — a four-year span from licensing to showroom, and a reminder that even with organizational commitment, novel architectures rarely ship on aggressive schedules. Development was led by Forrest McFarland, Packard’s Chief Engineer in Advanced Engineering, whose team included a young engineer named John Z. DeLorean. McFarland later documented the engineering in SAE Technical Paper 560026, “The New Packard Torsion Level Suspension.”

Competitors took notice. Chrysler, which had planned to introduce torsion front suspension, delayed its rollout a year rather than compete with Packard’s more elaborate implementation on introduction.

Integrated Control: Sensing and Actuation

Packard layered an active element atop the passive mechanical design. A level sensor and a single electric motor adjusted the preload on a pair of auxiliary short torsion bars, correcting chassis attitude for static load distribution without changing the nominal ride height. The system responded to sustained imbalance with a deliberate delay — several seconds — to avoid chasing transient road inputs. It is an early production example of what would today be called a closed-loop mechatronic control function on an automobile, implemented with 1950s electromechanical hardware.

SE Lessons for Practitioners

- **Architecture as a lever on emergent behavior.** Pitch, roll, and squat are not attributes of any single component; they emerge from how subsystems are coupled to the chassis. Torsion-Level demonstrates that decoupling is not always the right answer — sometimes deliberate mechanical coupling is the architectural move that collapses a trade space.

- **Passive plus active, in the right order.** The mechanical design delivered the core dynamic benefits; the electrical leveling system addressed a separate quasi-static requirement. Layering an active control loop atop a sound passive architecture — rather than asking control to compensate for a deficient mechanical design — remains good practice in mechatronic systems today.

- **Technology transfer requires an integrator.** Allison's concept existed for years before finding an organization with the resources and architectural freedom to develop it into a production system. Inventors produce concepts; integrators produce products.
- **Technical excellence does not guarantee program survival.** By 1956, Packard production in Detroit had ceased; the final Detroit-built Packard rolled off the line on June 25, 1956. The 1954 Studebaker merger (in which Packard acquired the financially distressed Studebaker), the cramped Conner Avenue body plant, and eroding dealer confidence overwhelmed every engineering advantage Packard could deliver. Systems engineers who optimize only the technical system — ignoring manufacturing, supply chain, financial, and market subsystems — risk shipping a beautifully engineered product into an enterprise that cannot sustain it.
- **Durability assumptions deserve scrutiny**. Seven decades on, surviving Torsion-Level cars commonly exhibit compressed bushings, reduced ride height, and compensator units that adjust in only one direction. These symptoms trace to design assumptions about bushing resilience and motor duty cycle that were reasonable for a five-to-ten-year service life but have not held over decades of continued use. Long-life assumptions baked into novel mechanisms should be validated against realistic service profiles, not asserted.

Sidebar: Two Switches, Two Use Cases

Torsion-Level Ride came equipped with a factory cutoff switch — a feature not common among competitors' early automatic level control systems, most of which were wired through the ignition-switch RUN position. Packard's engineers went a step further late in the 1956 model year, offering a dealer-installed manual override switch that let the driver command the compensator motor up or down directly.

A story widely repeated in hobby circles holds that children bouncing on the rear bumpers of parked Packards could run the battery down as the leveler faithfully pumped the car back to attitude. The primer author at Packard Motor Car Information is skeptical — the factory cutoff mitigated exactly this scenario for any owner who used it. Whether the battery-drain problem was widespread or mostly apocryphal, the defensive feature was baked into the design.

The override switch enabled a more interesting use case. Rather than waiting for the automatic system to respond to a weight shift, the owner could drive the motor directly: raise the rear end well above its design attitude, then change a tire with only a few inches of jack travel required. As one long-time Packard owner summarized on the forum, the override "is also very handy when changing a rear tire."

The override circuit still ran through the system's limit switches, preventing the compensator from being driven past its mechanical extremes regardless of how the driver held the switch — a discipline worth preserving in any modern control design.

Additional SE Lessons

- **Protective controls get repurposed.** A cutoff originally intended as a safety and service feature served in the field as a theft-of-service mitigation. Multi-purpose protective functions are often discovered after deployment — an argument for making them accessible and well-documented rather than buried.
- **Actuators are capabilities, not just functions.** The compensator motor was specified to perform static load leveling. Once the manual override made it a commandable actuator, it became a small electric lift for roadside tire changes. Any sufficiently powerful actuator will eventually be used for something outside its requirements baseline — a pattern worth anticipating in hazard analysis and in owner documentation.

- **Interlocks travel with the function.** Preserving safety interlocks across every control path, whether automatic or manual, is as relevant to drive-by-wire vehicles and industrial robotics today as it was to a 1956 Patrician.

Torsion-Level Ride was, by most contemporary and retrospective accounts, the most sophisticated production automotive suspension of its era. That it disappeared with the company that built it is not an indictment of the engineering — it is a reminder that systems engineering accountability extends past the product boundary.

References

McFarland, F., “The New Packard Torsion Level Suspension,” SAE Technical Paper 560026, 1956.

“Torsion-Level — A Primer,” Packard Motor Car Information: https://packardinfo.com/xoops/html/modules/newbb/viewtopic.php?topic_id=5493

“T/L Manual Control Switch,” Packard Motor Car Information forum: https://packardinfo.com/xoops/html/modules/newbb/viewtopic.php?viewmode=flat&topic_id=1326&forum=3

Mac’s Motor City Garage, “Packard Torsion-Level Ride — How It Works”: https://macsmotorcitygarage.com/video-packard-torsion-level-ide-how-it-works/

Mac’s Motor City Garage, “A Deep Dive Into Packard’s Torsion-Level Suspension”: https://macsmotorcitygarage.com/video-a-deep-dive-into-packards-torsion-level-suspension/

Hagerty, “The suspension on the Packard Caribbean was a new twist for 1955”: https://www.hagerty.com/media/video/the-suspension-on-the-packard-caribbean-was-a-new-twist-for-1955/

HotCars, “Packard’s Amazing Electric Torsion-Level Suspension”: https://www.hotcars.com/packards-amazing-torsion-level-suspension/

Packard Motor Car Information, sales brochure, “How Packard Torsion-Level Suspension Gives You The Smooth Safe Ride You’re Now Enjoying”: https://packardinfo.com/xoops/html/downloads/SmoothSafeRide.pdf

Motor Trend, road test of Packard Clipper with Torsion-Level Ride, June 1955.

This article was drafted with assistance from Claude.ai by Anthropic.

This Month in SE History: Special Double Feature

Sixty-Five Years Ago: Kennedy’s Challenge and the Power of a Well-Formed Requirement

May 25, 1961

On May 25, 1961, President John F. Kennedy stood before a joint session of Congress and delivered his Special Message on Urgent National Needs—a sweeping address covering foreign aid, civil defense, and other Cold War priorities. Near the midpoint of the speech, he issued what may be the most consequential top-level requirement in the history of systems engineering: “First, I believe that this nation should commit itself to achieving the goal, before this decade is out, of landing a man on the moon and returning him safely to the Earth.” Sixty-five years later, Kennedy’s sentence remains a textbook example of how to frame an objective so that an engineering enterprise can actually execute against it.

The Context: A Program Without a Purpose

Kennedy inherited a space program that was visibly trailing the Soviets. Sputnik had flown in October 1957. Yuri Gagarin had orbited the Earth on April 12, 1961. The Bay of Pigs operation had collapsed five days later. Alan Shepard's suborbital Freedom 7 flight on May 5, 1961, had lasted fifteen minutes and reached an apogee of roughly 116 miles—a fraction of Gagarin's orbital achievement. Kennedy had asked Vice President Lyndon Johnson, as chairman of the National Space Council, to identify a space goal the United States could plausibly reach before the Soviet Union. Johnson's consultations with NASA Administrator James Webb, Wernher von Braun, and others converged on a crewed lunar landing as the first objective where Soviet booster superiority could be offset by a sufficiently ambitious American program.

Anatomy of the Requirement

Read as a requirement statement, Kennedy's sentence is remarkable for what it contains and what it deliberately omits. It is:

Unambiguous. "A man on the moon" and "returning him safely to the Earth" leave no room for reinterpretation. Orbiting the Moon, impacting the Moon, or one-way missions are all excluded.

Verifiable. Success is binary and publicly observable. There is no ambiguous acceptance criterion.

Time-bounded. "Before this decade is out" sets a hard deadline of December 31, 1969—roughly eight years and seven months from the speech.

Safety-constrained at the top level. The word "safely" elevates crew survival from a derived requirement to a primary objective. The architecture must return the crew; a mission that lands but strands the crew is a failure.

Architecture-agnostic. Kennedy specified the what, not the how. The speech explicitly acknowledged open trades: "we propose to develop alternate liquid and solid fuel boosters… until certain which is superior." The lunar mission mode decision—direct ascent versus Earth-orbit rendezvous versus lunar-orbit rendezvous—was not resolved until mid-1962, more than a year after the commitment was made.

The SE Lesson: Commit Resources Before the Architecture Is Known

Kennedy asked Congress for an initial $531 million in FY1962 and warned that the program would cost $7 to $9 billion over five years—figures committed before the Apollo architecture was selected, before the Saturn V was designed in detail, and before a lunar landing had been demonstrated in any form. This runs counter to a common instinct to defer resource commitments until design maturity is high. The Apollo counterexample suggests that when a top-level requirement is sufficiently clear, verifiable, and time-bounded, it can pull an architecture into existence on schedule. A vague requirement, no matter how well-funded, tends to produce drift.

Neil Armstrong and Buzz Aldrin landed in the Sea of Tranquility on July 20, 1969, and returned to Earth on July 24, 1969—five months inside Kennedy's deadline. Kennedy did not live to see the program's execution, and the political coalition that supported Apollo eroded almost as soon as the objective was met. But the engineering enterprise delivered. The lesson for practicing systems engineers is not that national commitment makes impossible things possible; it is that a properly formed top-level requirement is a force multiplier for everything downstream.

A necessary caveat: a schedule-driven top-level requirement also carries risk. The pressure to meet Kennedy's deadline was a contributing factor in the Apollo 1 fire of January 27, 1967, which killed astronauts Grissom, White, and Chaffee during a launch pad test. The subsequent investigation revealed workmanship and design deficiencies that had accumulated under schedule pressure. Apollo recovered—the program stood down for twenty months, redesigned the command module hatch and wiring, and ultimately met the deadline—but the episode is a reminder that a well-formed requirement does not exempt a program from the consequences of compressing margin in execution.

Fifty-Five Years Ago: Mariner 9 and the Value of Reprogrammable Margin

May 30, 1971

On May 30, 1971, an Atlas-Centaur lifted off from Launch Complex 36B at Cape Kennedy carrying Mariner 9. One hundred and sixty-seven days later, on November 14, 1971 (UTC), Mariner 9 fired its main engine for 915.6 seconds and entered orbit around Mars—the first human-built object to orbit another planet. What happened between arrival and the start of useful science operations is the real systems engineering story, and it remains instructive fifty-five years on.

A Mission Rebuilt from a Failure

Mariner 9 was not designed to fly alone. The Mariner Mars 71 project was a two-spacecraft mission: Mariner 8 was tasked with mapping 70 percent of the Martian surface while Mariner 9 studied temporal variations in atmosphere and surface features. The paired-launch architecture, used by every preceding Mariner project, provided redundancy against launch vehicle and early-cruise failures.

That redundancy was tested immediately. On May 8, 1971, Mariner 8 lifted off on an Atlas-Centaur and was lost when a failed integrated circuit in the Centaur's pitch guidance electronics caused the upper stage to tumble. The spacecraft fell into the Atlantic Ocean roughly 400 kilometers north of Puerto Rico. NASA had three weeks to decide whether to fly Mariner 9 with its original narrow objectives or rewrite the mission to absorb Mariner 8's mapping responsibilities as well. The team chose the latter, reprogramming the surviving spacecraft's observation plan while investigators were still confirming that the Centaur failure was not also a threat to Mariner 9. The fact that this was possible at all—on a three-week timeline, against a hard planetary launch window—reflects both disciplined configuration management and a spacecraft architecture that treated its observation sequences as data rather than hardware.

Arrival into a Planet-Wide Dust Storm

Mars had other plans. In late September 1971, a regional dust storm began in the Noachis region and grew into what was, at the time, the largest global dust storm ever observed on the planet. By the time Mariner 9 arrived in mid-November, the surface was almost entirely obscured; only the summits of Olympus Mons and the three Tharsis volcanoes were visible above the dust. The carefully planned mapping sequence—the one the team had rewritten three weeks before launch—was now pointed at a featureless orange haze.

Two Soviet spacecraft arrived at Mars during the same storm. Mars 2 (launched May 19, 1971) and Mars 3 (launched May 28, 1971) had far less capacity to alter their observation plans from Earth, and largely executed their preprogrammed sequences as designed. They returned limited useful surface data and exhausted consumables against an obscured planet. Mars 3 delivered a lander that transmitted for roughly 14.5 seconds before failing, possibly due to the same storm.

Mariner 9 was reprogrammed from Earth. Ground controllers at JPL commanded the spacecraft to suspend surface imaging and enter a conservation mode, preserving tape recorder cycles and attitude control gas for use after the storm abated. Imaging resumed in mid-January 1972. Over the following 349 days in orbit, Mariner 9 returned 7,329 images covering approximately 85 percent of the Martian surface, along with the first close-up images of Phobos and Deimos. The mission ended on October 27, 1972, when the spacecraft exhausted its gaseous nitrogen attitude-control supply.

The SE Lessons

Mariner 9's success is frequently and correctly attributed to reprogrammability, but the lesson is broader than "make the software updatable." Three distinct principles are worth extracting:

Flexibility is a form of margin. Margin is traditionally tracked in mass, power, propellant, and schedule. Mariner 9 demonstrated that the ability to modify operational plans in flight is itself a reserve against unforeseen environmental conditions. The Soviet probes had comparable hardware capability; what they lacked was headroom in their command architecture.

Paired-launch redundancy solves for the launch phase, not the mission. The Mariner 8/9 pairing protected against launch vehicle failure and it worked as intended. What it could not protect against was the loss of mission objectives the second spacecraft would have covered. Programs that depend on paired assets should explicitly analyze degraded-mission scenarios and pre-plan which objectives survive a single-asset failure.

Consumables, not components, often define end-of-life. Mariner 9's mission did not end because of a component failure; it ended when attitude-control nitrogen ran out. Fifty-five years later, this is still the dominant end-of-life mode for long-duration spacecraft, and it reinforces the value of operational modes—like Mariner 9's dust-storm safing—that can stretch consumables when the environment demands it.

A Cross-Cutting Observation

The JFK and Mariner 9 anniversaries bookend a specific idea: that system success depends at least as much on the quality of the top-level framing and the preservation of adaptive headroom as it does on the technical excellence of any individual subsystem. Kennedy's requirement was unambiguous enough to drive an unprecedented engineering program to completion inside a decade. Mariner 9's designers left enough room in their operational architecture to rewrite the mission twice—once on the ground before launch, once in flight at Mars. Both outcomes are reminders that systems engineering is not only the discipline of getting the details right, but also the discipline of leaving the details changeable for as long as possible.

References

- JFK Library, "Address to Joint Session of Congress May 25, 1961": https://www.jfklibrary.org/learn/about-jfk/historic-speeches/address-to-joint-session-of-congress-may-25-1961
- NASA, "The Decision to Go to the Moon": www.nasa.gov/history/the-decision-to-go-to-the-moon
- NASA Science, "Mariner 9": https://science.nasa.gov/mission/mariner-9
- Mariner 9 – Wikipedia: https://en.wikipedia.org/wiki/Mariner_9

This article was drafted with assistance from Claude.ai by Anthropic.

INCOSE Michigan Chapter

www.incose.org/michigan | Follow us on LinkedIn

Fostering a world-class systems engineering environment

Portions of this newsletter were researched and drafted with assistance from Claude.ai (Anthropic, 2026).