2026 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
***DIGITAL ENGINEERING / SYSTEMS ENGINEERING* TECHNICAL SESSION**
AUGUST 11-13, 2026 - NOVI, MICHIGAN

DECISION ENGINEERING: APPLYING LEAN PRODUCTION PRINCIPLES TO THE LIFECYCLE DECISION FACTORY

Eric Alexander, CSEP[1], Dr. Matthew Foglesong, PhD[1], Louis (Bill) Berklich[2]
[1,]Sabel Systems Technology Solutions, LLC
[2]U.S. Army DEVCOM Ground Maneuver Center (GMC)

ABSTRACT

The Army's transformation mandate is unambiguous: deliver warfighting capability 25–30% faster. Every Tier 2 metric published in support of that mandate – days between milestones, days to complete the requirements process, days to complete contracting, days to complete testing – is really a decision throughput measurement. Yet the systems engineering (SE) discipline that governs those timelines has no formal production framework for producing decisions. This paper proposes Decision Engineering as the framework. Grounded in lean production theory, applied to information work and anchored to the defense acquisition policy structure of DoDD 5000.01 and DoDI 5000.02, Decision Engineering reconceives SE as the discipline of designing, operating, and continuously improving the lifecycle decision production system. It introduces a formal decision ontology comprising six decision states (Latent, Declared, Active, Deferred, Closed, Reopened), four topology relationship types (Precedes, Enables, Constrains, Triggers), and five diagnostic biomarkers that measure production system health. The paper presents the ***Decision Factory*** *model and describes how its application to ground vehicle system programs and across the Army acquisition enterprise can accelerate capability delivery without sacrificing rigor. Defense acquisition programs do not fail for lack of engineering rigor or digital tooling. They fail because the production system that converts information into decisions is ungoverned – no inventory count, no flow discipline, no throughput measurement. Decision Engineering names this production system, provides instruments to measure its health, and offers a practitioner framework for running it better.*

Citation: E. Alexander, M. Foglesong, L. Berklich, "Decision Engineering: Applying Lean Production Principles to the Lifecycle Decision Factory," Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium *(GVSETS), NDIA, Novi, MI, Aug. 11*–13, 2026.

1. INTRODUCTION: WHAT SYSTEMS ENGINEERING IS ACTUALLY PRODUCING

Ask any experienced systems engineer what the SE organization produces and the answer is consistent: models, specifications, architectures, interface control documents, trade study reports, systems engineering plans (SEP). These are the measurable outputs – the deliverables that milestone reviews assess, that contracts specify, and that Contractor Performance Assessment Reporting System (CPARS) ratings are built on.

Ask what any of those artifacts is for, and the answer is equally consistent: to inform a

DISTRIBUTION A. Approved for public release; distribution unlimited. OPSEC #: 10504 decision. The architecture model exists to support an architecture selection decision. The interface control document (ICD) exists to document an agreed interface standard. The trade study exists to recommend a course of action.

The SEP exists to document how the program has decided to manage its technical activities. which raises a question the profession has not formally confronted: *when is a model, a specification, or an ICD actually done?* Not contractually delivered – actually done, in the sense of having accomplished what it was designed to accomplish.

The answer, pursued rigorously, is that an artifact is done when the decision it exists to inform has been made – closed by the right authority, with the right information, with its rationale documented and its downstream dependencies notified. Until that moment, the artifact is a work in progress. Until that point the artifact is inventory accumulating in the SE production pipeline. It has consumed resources without producing the output it was built to enable.

Systems engineers are in the decision production business. The artifacts are the packaging. The decisions are the product. This reframing is not semantic – it has direct structural consequences for how programs are governed, how governance is measured, and why programs with sophisticated digital engineering infrastructure continue to fail at milestones in predictable ways.

1.1 The Policy Mandate Already Knows This

DoDD 5000.01 directs the Defense Acquisition System to deliver performance at the speed of relevance and enumerates the operating policies needed to achieve that speed: empower program managers, simplify acquisition policy, employ tailored approaches, conduct data-driven analysis, actively manage risk [1]. Every one of those policies is a decision governance instruction. Empowering the PM means giving them authority to close decisions without unnecessary escalation. Simplifying policy means reducing the overhead that slows decision flow. Tailoring means calibrating process intensity to actual decision consequence. Data-driven analysis means making decisions with information rather than by default. Active risk management means declaring the decisions that are being deliberately deferred and owning the exposure.

The Army Transformation Objectives make the quantitative case explicit: a 25–30% increase in the speed of delivery, measured through Tier 2 metrics that include days between milestone events, days to complete requirements, contracting, and testing [2]. These are not schedule metrics. They are cycle time measurements of a decision production system. The days between Milestone B and Milestone C are the cumulative processing time of every decision that had to be made in Engineering and Manufacturing Development. Compressing that timeline requires improving the throughput of the decision production system, not simply applying more schedule pressure to a system with no flow discipline.

1.2 The Digital Engineering Gap

The past decade of defense acquisition transformation investment has produced significant modernization: model-based systems engineering adoption, DevSecOps pipelines, digital threads, cloud-hosted PLM environments, and enterprise analytics platforms. These investments modernized the artifact production system. However, they did not redesign the decision production system.

Programs with mature digital engineering stacks continue to experience multi-month Engineering Change Proposal (ECP) adjudication cycles, architecture churn entering the engineering and manufacturing development (EMD) phase, governance fatigue from review boards that compensate for upstream ambiguity rather than validating clean decisions, and sustainment risk discovered post-production at maximum correction cost. And while the digital tools are sophisticated, the decision pipeline is still ungoverned.

This paper addresses the current gap in the ungoverned SE decision pipeline. In this paper the theoretical foundation is established in Section 2 and utilizes lean production principles applied to information work. The formal decision ontology is introduced in Section 3. Section 4 presents the five diagnostic biomarkers and the Decision Energy framework. the Decision Energy framework and the five related diagnostic biomarkers are presented in Section 4. Section 5 describes the Decision Factory model and its application. And, Section 6 addresses the implications for ground vehicle system programs and the Army acquisition enterprise.

2. THEORETICAL FOUNDATION: LEAN PRODUCTION APPLIED TO DECISIONS

2.1 The Toyota Production System and Information Work

Taiichi Ohno's development of the Toyota Production System in the postwar period established production principles that have proven remarkably durable across manufacturing domains: eliminate waste from the production process, create continuous flow, use pull systems to avoid overproduction, level the production load, stop and fix problems at the origin, standardize tasks, build quality in rather than inspect it out, and establish a culture of continuous improvement [3].

Don Reinertsen extended these principles to product development in his 2009 work on product development flow, arguing that the economics of product development – the cost of queue delay, the value of fast feedback, the waste of batch processing – are structurally identical to manufacturing economics even though the work product is information rather than physical goods [4]. The insight that a decision sitting in a review queue for six weeks instead of being closed in one carries a measurable cost – the value of everything downstream that could not begin until the decision is closed, multiplied by the delay is a concept that applies with full force to defense acquisition programs.

This paper extends that lineage specifically to the defense acquisition decision production system, mapping each core TPS principle to a named dysfunction pattern and a named intervention.

2.2 The Production Analogy Applied to Systems Engineering

A production plant takes in raw material, applies people, processes, and tools, and produces finished goods. A defense acquisition program takes in raw information: stakeholder needs, threat updates, technology maturation reports, design defects, engineering change proposals, taskers from higher headquarters then applies systems engineers, SE processes, and digital tools, and should produce decisions. The decisions are the finished goods. The models and specifications are the packaging that communicates and implements the decisions downstream.

When this production frame is applied, the canonical dysfunction patterns of defense acquisition programs become immediately apparent as production system failures, with lean production root cause, These are:

Critical Design Review (CDR) Ambush - the 'technical surprise'

Escaped production defect:

- a latent decision that was never declared passed through all upstream reviews undetected, was incorporated into downstream work as an assumption, and surfaced at CDR when the cost of correction was maximized.

ECP Avalanche in EMD

Rework rate surge:

- decisions closed before sufficient information was available generated defective downstream work. Correction requires reopening closed decisions and cascading changes through the topology.

Compensatory Governance Spiral

Adding inspection stations to a backed-up pipeline:

- review boards are added in response to governance failures without addressing the root cause (undeclared decisions, fractured ontological alignment). The pipeline gets slower, not safer.

Re-litigation at every review

Batch processing failure:

- decisions are not closed at the level they are made. They recirculate through governance forums without resolution, consuming decision authority bandwidth without producing closed decisions.

Sustainment surprise post-production
Missing pull signal:

- sustainment, logistics, and operations data did not flow upstream to influence design decisions while the cost of change was still manageable.

Each of these failure modes has a structural intervention at the production system level. Naming the production system is the prerequisite for applying the interventions.

2.3 The Cost of Queue Delay in Decision Inventory

In lean production, work in progress sitting between production stations is not a neutral condition. It is waste: consuming space, requiring handling, obscuring quality problems, and preventing downstream stations from starting their work. The longer it sits, the more it costs.

Decision inventory is the acquisition equivalent. An unresolved interface question is not neutral. Every engineer on both subsystem teams is making assumptions to fill the gap. Some assumptions will align. The ones that don't align will produce integration defects that surface at CDR or later as carrying costs that compound with every week the question stayed open. Though the decision was not made the program did not stop. The decision inventory however, accumulated and that cost is being paid continually, invisibly, as program delay and rework potential.

Reinertsen's cost of delay framework [4] quantifies this: every decision on the critical path that sits in a queue for an additional week costs the value of whatever downstream work it blocks multiplied by the delay. For a program with a 36-month EMD phase and a schedule margin of six weeks, a single consequential decision that takes eight weeks instead of two to close can absorb the entire schedule margin — before any other delay has occurred.

3. DECISION ONTOLOGY: A FORMAL STRUCTURE FOR DECISION ENGINEERING

Decision Engineering requires a precise object model for the decision – not as a governance abstraction, but as a structured work product with defined internal anatomy, a formal lifecycle, measurable properties, and a topology connecting it to every other decision in the program. The decision object model fundamentals include developing a well formed decision, recognizing the available decision states, defining the decision topology and understanding the decision critical path.

3.1 The Five Elements of a Well-Formed Decision

Every well-formed decision contains five structural elements. When any element is absent, the decision cannot close cleanly – and apparent closure will reopen under program pressure.

1. **Question**

The bounded choice being resolved, stated as a specific question with a finite answer space. If the question cannot be stated in one sentence with discrete options, it has not been declared.

2. **Decision Space**

The bounded set of possible answers, or the alternatives under consideration. Without a declared decision space, decisions drift as new alternatives emerge.

3. **Forcing Function**

The specific event or deadline that makes resolution necessary. Without a forcing function, low-urgency decisions accumulate indefinitely.

4. **Resolution**

The chosen answer plus rationale. Rationale is what keeps the decision closed when new stakeholders arrive and want to re-litigate.

5. **Authority**

The named individual with organizational standing to close this decision and have it stay closed. Not a committee. One person, by name.

The most common reason decisions do not close is not that they are hard, it is that one or more of these elements is missing. The question is too broad. The decision space was never bounded. The authority was assumed rather than declared. Every re-litigation of a closed decision traces back to at least one missing element.

3.2 The Six Decision States

Every decision in a program exists in exactly one of six states. The state determines what management action is appropriate, what the carrying cost is, and the risk exposure it creates. These decision states are: latent declared, active, deferred, closed, and reopened. Programs that cannot report the state distribution of their decision inventory are operating the production system without gauges.

1. **LATENT**

Active and shaping outcomes but not yet recognized as a decision. No declaration, no owner, not in any register. This is the mechanism of the CDR ambush and the most dangerous state in the lifecycle.

2. **DECLARED**

Named, bounded, and entered into the decision register. A question has been stated, a decision space identified, an authority named, a forcing function assigned. No longer Latent.

3. **ACTIVE**

Analysis underway. Trade studies, modeling, stakeholder consultation in progress. Decision is in the pipeline. Cycle time accumulating.

4. **DEFERRED**

Deliberately postponed with documented rationale, declared risk exposure, a named revisit trigger, and a named deferral owner. Not the same as Latent.

5. **CLOSED**

Resolved. Answer chosen, rationale captured, authority confirmed, downstream notified. In historical record.

6. **REOPENED**

Returned to Active from Closed due to new information or downstream trigger event. Normal when managed; pathological when recurrent or authority-less.

The transition from Latent to Declared is the highest-value governance act in Decision Engineering. A program that systematically surfaces Latent decisions before CDR has fundamentally changed its risk profile – not by adding governance overhead, but by eliminating the source condition that makes compensatory governance necessary.

3.3 Decision Topology: The Relational Network

Decisions do not exist in isolation. They are connected in a dependency network whose topology determines the sequence in which they can be made, the results if delayed, the cascade effects when reopened, and the critical path for program throughput. Four fundamental relationship types govern the topology; precedes, enables, constrains, and triggers. each of these relations has a unique failure mode when unmanaged.

1. **PRECEDES**

Decision A must close before Decision B can be properly bounded. Sequencing dependency. Violation produces ECP cascade: B closes on assumptions about A that prove incorrect when A closes differently.

2. **ENABLES**

Decision A creates conditions for B to close with confidence. Information dependency. B can be declared and analyzed while A is active — but should not close until A closes. Violation produces the PDR-to-CDR baseline instability pattern.

3. **CONSTRAINS**

Resolution of A limits the answer space of B without fully determining it. Violation produces invalid trade study alternatives – options that are not viable given A's resolution, discovered after the trade study is complete.

4. **TRIGGERS**

When A reopens or changes state, B is automatically returned from Closed to Reopened. The cascade relationship. Invisible topology produces ECP avalanches when a root decision changes and its triggered descendants reopen without governance recognition.

3.4 The Decision Critical Path

Not all decisions govern program outcomes equally. The Decision Critical Path is the chain of decisions, connected by Precedes and Enables relationships, where each node has (1) a downstream reach, affecting three or more subsequent decisions, (2) an irreversibility cost, meaning incorrect closure cannot be undone without reopening multiple downstream decisions, and (3) a schedule sensitivity, meaning a delay in closure will delay work that cannot begin without it. Most programs at any given phase have five to twenty decisions on the critical path, out of the fifty to one hundred that exist in the register. Decision Engineering concentrates governance rigor on the critical path decisions and deliberately streamlines governance on decisions with limited downstream reach. This is not reduced rigor, it is calibrated rigor. The program that applies identical governance intensity to all decisions has not achieved thoroughness. It has achieved indiscrimination, which produces the governance overhead that makes programs slower without making them safer.

4. THE DECISION ENERGY FRAMEWORK AND FIVE BIOMARKERS

4.1 Decision Energy

Decision Engineering defines program health through a single integrating metric:

Decision Energy = Decision Velocity × Decision Coherence

Decision Velocity measures how quickly decisions close – from declaration to closure, averaged across the active decision register. Decision Coherence measures whether decisions close in structural alignment – whether the decision in architecture propagates correctly to the Work Breakdown Structure, cost model, logistics construct, and configuration baseline. Velocity without coherence produces instability: programs that appear to be moving fast but that re-litigate the same decisions at every review because downstream alignment was never established. Coherence without velocity produces stagnation: technically rigorous programs unable to close consequential choices at the speed the mission demands.

The product, *Decision Energy*, is what the Decision Factory works to increase. Unlike artifact volume, review board count, or milestone compliance rates, Decision Energy measures the actual health of the decision production system rather than its outputs. A program can have full milestone compliance and degraded Decision Energy simultaneously, which is precisely the condition that produces the CDR ambush.

4.2 The Five Biomarkers

Decision Energy is monitored through five diagnostic biomarkers: decision cycle time, architectural coherence, governance strain, ECP churn, and sustainment feedback. Each marker is individually diagnostic and collectively definitive and draw on the same observational logic as clinical biomarkers: a single elevated reading signals a specific subsystem under stress; multiple elevated readings indicate systemic degradation requiring structural intervention.

1. **Decision Cycle Time**:

Duration from declaration to authoritative closure. *Healthy:* decisions close in days to weeks. *Degraded:* ECP adjudication takes months, owners unclear, decisions relitigated across forums.

2. **Architectural Coherence:**

Alignment between stakeholder needs, requirements, architecture, WBS, cost structure, and sustainment artifacts. *Healthy*: single system definition referenced across all domains. *Degraded*: multiple competing definitions, constant manual bridging.

3. **Governance Strain**

Ratio of compensatory to additive governance. Additive governance validates clean decisions. Compensatory governance substitutes for decisions not yet made. *Healthy*: reviews validate, high closure rate. *Degraded*: reviews relitigate, boards compensate for absent decisions.

4. **ECP Churn Rate**

Rate at which engineering changes accumulate and are processed. High ECP churn is evidence of decisions closed before sufficient information was available – not a change management problem. *Healthy*: baseline stable, ECPs processed continuously. *Degraded*: ECP backlog accumulates, configuration drift visible.

5. **Sustainment Feedback Strength**

Degree to which operational and sustainment data actively shapes upstream design decisions. *Healthy*: field performance and depot data influence architecture trades. *Degraded*: sustainment risk discovered post-production, organic transition assumptions unvalidated.

4.3 The Ontological Spine

Underlying all five of the biomarkers is the Ontological Spine which is the structure or chain of alignment connecting the program's architecture definition to its Work Breakdown Structure, cost model, and configuration baseline. When the spine is intact, a change to any element propagates visibly through all others. When it fractures through undeclared decisions made under schedule pressure, through architecture updates not reflected in the WBS, through cost elements that trace to functions no longer in the architecture then multiple competing system definitions emerge and the governance system compensates manually for the structural gap.

The forensic test for spine integrity is to select any cost element in the program budget, . trace it to a function in the architecture, then trace that function to a work package in the WBS. If traceability fails at any step without relying on informal knowledge, the spine has fractured. The governance overhead visible in elevated Governance Strain and Architecture Coherence biomarkers is the cost of the fracture, not the cause of it. Repairing the fracture is the intervention. Adding review boards is the compensatory response that makes the fracture permanent.

5. THE DECISION FACTORY MODEL

5.1 Architecture

The Decision Factory is the operational framework that implements Decision Engineering principles within a defense acquisition program. It consists of three integrated components:

Component 1: Decision Register and State Machine. A structured register of all declared decisions, each with its five structural elements populated, its current state tracked, and its topology relationships mapped to precursor and dependent decisions. The register is the program's production floor inventory count – the instrument that makes Latent decisions visible by requiring that every consequence-bearing question be declared.

Component 2: Dynamic Rigor Calibration. A rules-based mechanism that assigns governance rigor intensity to each decision based on its properties: phase of the lifecycle, downstream reach in the topology, reversibility cost, safety criticality, MOSA/interface consequence, and data rights exposure. High-consequence decisions receive intensive scrutiny — formal trade studies, documented authority, structured review, rationale capture. Decisions with limited downstream reach are deliberately streamlined. The calibration is deterministic, not personality-driven, and it operationalizes the AAF tailoring mandate at the decision level rather than the program level.

Component 3: Biomarker Telemetry and Throughput Measurement. Continuous measurement of the five biomarkers – Decision Cycle Time, Architectural Coherence, Governance Strain, ECP Churn Rate, and Sustainment Feedback Strength – against baseline and trend. When biomarkers degrade, the DECISION FACTORY identifies the structural intervention rather than the compensatory governance response. Elevated Governance Strain calls for upstream structural clarity – not more review boards. Elevated ECP Churn calls for decision sequencing correction – not faster change processing.

5.2 Applying DECISION FACTORY: The Phase-by-Phase Decision Registry

Decision Engineering is a lifecycle discipline. Every acquisition phase produces a distinct decision domain with phase-specific critical path characteristics. The Decision Factory instruments all of them.

Lifecycle Phase	Primary Decision Domains	Dominant Biomarker(s)

Concept / Pre-MDD	Mission need sufficiency; technology maturation pathway; make/buy/mature; MOSA framing; affordability bounds; materiel solution viability	Architectural Coherence – ontological alignment established early determines structural integrity for all downstream phases
TMRR	Requirements stability; architecture selection; COTS integration strategy; prototyping approach; verification strategy; interface control baseline	Decision Cycle Time; Architectural Coherence; Governance Strain – highest-density decision phase with maximum downstream consequence
EMD	ECP adjudication; PDR/CDR exit criteria sufficiency; test anomaly disposition; configuration baseline integrity; production readiness sequencing; GFE/GFI coordination	ECP Churn Rate; Decision Cycle Time – most expensive phase to absorb decision churn; critical path compression here drives most of the 25–30% speed improvement
Production & Deployment	Configuration control; lot acceptance criteria; production anomaly disposition; fielding sequence prioritization; organic sustainment transition readiness	ECP Churn Rate; Sustainment Feedback Strength
Operations & Support	Obsolescence management; PIP/ECP adjudication; tech refresh sequencing; sustainment contractor performance; organic capability investment	Sustainment Feedback Strength; Decision Cycle Time – most expensive phase by lifecycle cost but systematically undergoverned from a decision standpoint

5.3 The Deferred Decision Standard

One of the most important Decision Factory contributions is the formal treatment of deferred decisions. Deferral has a negative reputation in acquisition governance because it is associated with procrastination, indecision, or political avoidance. But there are decisions that should be deferred such as a sustainment data package depth decision that cannot be made until production quantity is confirmed; a software containerization decision that should wait for an Authority To Operate (ATO) assessment; or an interface standard decision appropriately deferred pending a technology maturation result.

The distinction between legitimate and negligent deferral is documentation. A legitimately deferred decision carries: (1) a rationale for why the deferral is appropriate and what information is unavailable; (2) a declared risk exposure identifying what downstream work proceeds on assumed answers; (3) a named revisit trigger – a specific event that returns the decision to Active state; and (4) a named deferral owner responsible for monitoring the trigger.

When this deferral standard is applied, the deferred decision becomes a precision governance instrument rather than a governance failure. Leadership can examine the deferred decision register and understand exactly what the program has chosen not to decide yet, at what risk, and who owns the trigger. Undeclared deferrals – decisions neither made nor declared deferred, simply floating in the program's ambient uncertainty – are not deferred governance, it is latent governance, which is the same mechanism as the CDR ambush operating at a longer time scale.

APPLICATION TO ARMY GROUND MANUEVER SYSTEM PROGRAMS

6.1 The Ground Vehicle Program Context

Army ground vehicle programs operate under a confluence of pressures that makes Decision Engineering particularly applicable. Modern ground platforms are complex, multi-vendor integration programs with complex system-of-systems integration requirements, mixed hardware/software stacks, and Government-responsible SoS integration against Army network and C2 architectures. They operate under sustained schedule pressure from Army transformation objectives while simultaneously managing Modular Open Systems Approach (MOSA) compliance requirements across four architectural layers: product architecture, mission systems architecture, digital engineering ecosystem, and organizational architecture.

This combination opens the full spectrum of potential Decision Factory failure modes. Technology Maturity and Resk Reduction (TMRR) phase architecture decisions made without MOSA governance produce EMD-phase ECP avalanches when the modular decomposition proves incorrect. Interface control decisions deferred without formal declaration produce CDR findings when vendor proprietary formats produce integration incompatibilities that no team declared as a decision. Digital engineering investments that modernize artifact production without redesigning decision governance produce sophisticated tooling environments and the same governance pathologies, now executed faster in better software.

6.2 MOSA as a Decision Timing Problem

Modular Open Systems Approach is widely understood as a product architecture requirement. The intent is to decompose the system into modules with open interfaces, maintain competitive access to hardware and software components, and preserve upgrade capacity to reduce sustainment costs. This understanding is correct but incomplete. **MOSA failure at the program level is almost always a decision timing problem, not a technical problem.**

By example, the power architecture decision that determines upgrade capacity for the next thirty years is often made by an engineer running a power budget model, without the decision having been declared to governance, without the downstream MOSA consequences having been mapped, and without a decision authority having formally committed to the choice. The interface standard selection decision is made by whoever populated the ICD under schedule pressure, without a formal decision record connecting the standard to an authority, a rationale, and a downstream notification. The data rights posture decision — perhaps the most consequential MOSA decision for organic sustainment and future competition, is frequently made by default, embedded in contract language drafted without explicit governance direction.

The Decision Factory, when applied to MOSA governance, changes this dynamic. Each of the MOSA-consequential decisions at each architectural layer is identified, declared, and placed on the decision critical path. The authority for each is named and the downstream dependencies are mapped. The timing is governed: not too early (before sufficient information exists), not too late (after downstream work has locked in consequences that require rework to correct).

6.3 Connecting to Army Transformation Metrics

The Army's Tier 2 speed metrics — days between milestones, days to complete requirements, contracting, and testing — directly measure the cycle time of specific decision domains within the acquisition lifecycle. The requirements process cycle time is the time required to declare, analyze, and close the decision set that constitutes an approved requirements baseline. The contracting cycle is the time required to close the decision set that constitutes a source selection. The testing cycle time is the time required to close the verification and disposition decisions that constitutes a test complete condition.

Each of these domains has a Decision Factory intervention. The Decision Cycle Time biomarker measurement establishes the baseline. The Latent decision identification removes the hidden inventory that pads cycle time without producing governance value. And the Decision Critical Path analysis identifies the small set of decisions that are actually driving the timeline. This in turn allows concentration of governance energy where it can be most effective. Employing Dynamic rigor calibration removes governance burden from decisions with low downstream consequence, creating moe bandwidth for intensive governance where it matters.

Decision Engineering does not achieve the Army's 25–30% speed improvement by compressing governance. It achieves it by eliminating the governance waste – the compensatory oversight, the re-litigation cycles, the escaped defects that surface downstream at maximum correction cost – that makes programs slow in the first place.

7. IMPLICATIONS AND CONCLUSION

7.1 For Systems Engineering Practice

Decision Engineering does not replace systems engineering. It names the governing discipline that systems engineering has always been working toward but has never formally defined. Every SE method whether MBSE, lean SE, model-based acquisition, digital engineering, is an attempt to help programs close decisions faster, with better information, at the right level of authority. Decision Engineering provides the production framework that gives those methods their purpose and the instruments that measure their actual contribution to program outcomes.

The practitioner identity shift implied is significant. The *Digital Systems Engineer* practices SE in a digitally enabled environment. The *Decision Systems Engineer* engineers the decision production system – designs it, instruments it, measures its health, and intervenes structurally when it degrades. The second role is not a replacement for the first. It is a governing layer above it that gives digital engineering investment its strategic purpose and its accountability measure.

7.2 For Program Office Leadership

The program manager whose chief engineer can answer the following three questions on demand is running a faster program: 1) Which decisions are currently open and who owns each? 2) Which are on the decision critical path? 3) Which decisions are currently Latent and shaping outcomes without having been declared? The ability to answer those questions is not a function of organizational talent. It is a function of instrumentation. Decision Factory provides the instrumentation.

The implication for program review structure is direct: every agenda item in every review board should be traceable to a specific decision that the review exists to close. If an agenda item cannot be connected to a decision closure event, it belongs in asynchronous information exchange, not in a decision authority's calendar. Decision authorities' time is the program's scarcest governance resource. The decision critical path is the filter for how it gets allocated.

7.3 Conclusion

The Army's commitment to delivering warfighting capability 25–30% faster is a decision throughput problem. The Tier 2 metrics that measure progress toward that commitment are decision cycle time measurements. The failure modes that currently consume the schedule margin those metrics are trying to recover — CDR ambushes, ECP avalanches, compensatory governance spirals, sustainment surprise — are all production system failures with structural interventions available.

Decision Engineering provides the framework for those interventions. It names the production system that systems engineering is actually operating, provides a formal ontology for the work product that system produces, gives practitioners instruments to measure production system health in real time, and establishes the practitioner identity — the Decision Systems Engineer — who is accountable for running the factory well.

The field has been practicing the artifact production system for decades. The policy — speed of relevance, empowered PMs, tailored rigor, data-driven decisions — has been describing the decision production system all along. Decision Engineering is *finally naming the thing the policy has been asking for*, and providing the discipline to deliver it.

REFERENCES

[1] DoD Directive 5000.01, "The Defense Acquisition System," September 9, 2020 (Change 1, July 28, 2022). Office of the Under Secretary of Defense for Acquisition and Sustainment.
[2] U.S. Army Acquisition Reform Measures of Success (Unclassified), Army Transformation Objectives and Tier 2 Metrics. Distribution A: Approved for Public Release.
[3] Ohno, T. (1988). Toyota Production System: Beyond Large-Scale Production. Productivity Press. (Original Japanese edition 1978.)
[4] Reinertsen, D.G. (2009). The Principles of Product Development Flow: Second Generation Lean Product Development. Celeritas Publishing.
[5] DoD Instruction 5000.02, "Operation of the Adaptive Acquisition Framework," January 23, 2020 (Change 1, June 8, 2022). Office of the Under Secretary of Defense for Acquisition and Sustainment.
[6] DoD Instruction 5000.88, "Engineering of Defense Systems," November 18, 2020. Office of the Under Secretary of Defense for Research and Engineering.
[7] IEEE Std 15288.2-2014, "Standard for Technical Reviews and Audits on Defense Programs." IEEE.
[8] DASD(SE) Best Practice 15288, "Best Practices for Contract Application of EIA-632 / ISO/IEC/IEEE 15288 on Defense Programs," Deputy Assistant Secretary of Defense for Systems Engineering, 2017.
[9] DoD Digital Engineering Strategy, "Implementing Digital Engineering across the Defense Acquisition Enterprise," June 2018. Office of the Deputy Assistant Secretary of Defense for Systems Engineering.
[10] Eppinger, S.D., and Browning, T.R. (2012). Design Structure Matrix Methods and Applications. MIT Press. (DSM methodology for dependency modeling adapted to decision topology application in this paper.)

ACRONYMS

Acronym	Definition
AAF	Adaptive Acquisition Framework
CDR	Critical Design Review
COTS	Commercial Off-The-Shelf
DE	Decision Energy (Velocity × Coherence)
DoDD	Department of Defense Directive
DoDI	Department of Defense Instruction
ECP	Engineering Change Proposal
EMD	Engineering and Manufacturing Development
GVSC	Ground Vehicle Systems Center
MBSE	Model-Based Systems Engineering
MDD	Materiel Development Decision
MOSA	Modular Open Systems Architecture
PDR	Preliminary Design Review

PLM	Product Lifecycle Management
PM	Program Manager
SE	Systems Engineering / Systems Engineer
SoS	System of Systems
SRR	System Requirements Review
TMRR	Technology Maturation and Risk Reduction
TPS	Toyota Production System
WBS	Work Breakdown Structure