

Reforming the Defense Equipment Technical Management System Driven by MBSE and the Digital Thread

Dan Yao^{1,*}, Junxiao Sun², Zhenyu Lei², Yiran Ma², Feng Hu¹, Shuangjiang Li³

¹ China Shipbuilding (Chongqing) Southwest Equipment Research Institute Co., Ltd, Chongqing 401122, China

² China Shipbuilding (Chongqing) Equipment Technology Co., Ltd, Chongqing 401122, China

³ CISDI Information Technology (Chongqing) Co.,Ltd, Chongqing 401122, China

*Corresponding Author.

Abstract

Document-centric technical management shows systemic bottlenecks for complex equipment across the life cycle, including unstable requirements, frequent interface changes, configuration drift, and weak quality closures. This paper proposes a five-element architecture-Process, Organization, Data, Tools, and Metrics-driven by MBSE (Model-Based Systems Engineering) and the Digital Thread, and builds a governance framework coupled to SRR/PDR/CDR/TRR, the three baselines (functional/allocation/product), and TRL/MRL. We elaborate three key mechanisms-configuration and change control, requirements traceability and interface management, reliability growth and FRACAS closure-and provide implementation guidelines on top-level design and governance, toolchain and data management, and staged pilots. Using illustrative simulated data, we demonstrate KPI improvement paths and dashboard styles to show operability. The approach can improve requirements stability and configuration conformance, reduce interface change rate and late-phase defects, shorten lead time, and lay the foundation for in-service data and digital twins as well as AI-enabled requirement quality assessment, defect prediction, and decision support.

Keywords

MBSE; Digital Thread; Systems Engineering; Configuration Management; Requirements Traceability; Reliability Growth; FRACAS; TRL/MRL.

1. Introduction

Over the past decade, aerospace, defense, energy, and high-end manufacturing have shifted from document-centric SE to MBSE. The INCOSE Systems Engineering Handbook positions the model as the authoritative source of truth for collaboration, verification, and delivery [1]. ISO/IEC/IEEE 15288 formalizes the interfaces among requirements, architecture, integration, and verification across the life cycle [2]. The U.S. DoD Digital Engineering Strategy promotes an Authoritative Source of Truth and a digital thread from requirements to sustainment, replacing multi-version documents with data and models [3]. NASA couples SRR/PDR/CDR/TRR with evidence packages and requires model consistency and verification traceability before each milestone [4].

For maturity management, the GAO TRA Guide and the DoD MRL Deskbook establish a two-dimensional TRL/MRL framework and emphasize a measurement-evidence-audit loop [6-7]. Reliability and safety standards such as GEIA-STD-0009 and IEC 61508 call for early reliability engineering and functional safety analysis with quantitative tracking of growth and risk reduction [8-9]. Configuration management standards MIL-HDBK-61A and ISO 10007 define

end-to-end processes from identification to change control, status accounting, and audits [12][15].

In China, GJB 9001C introduces process approach and risk-based thinking into defense quality management [10], while GJB 5000A specifies software capability and process maturity requirements [11]. With rising system complexity, industry practice is building MBSE modeling capability, integrating PLM/ALM, simulation management, and test data management, and piloting requirements traceability, interface control (ICD), and FRACAS. However, three challenges persist: multi-source requirement and interface changes, uncontrolled configuration/version proliferation, and weak measurement–risk–quality closure [1–15].

This paper contributes a unified governance solution that fuses MBSE and the digital thread into a five-element architecture; maps SRR/PDR/CDR/TRR to the three baselines and TRL/MRL; institutionalizes the gate admission logic of model consistency, metric thresholds, evidence packages, and audit trails; and constructs a data-driven closure around CCB, ICD/DSM, and FRACAS. We also provide KPI definitions and dashboards with illustrative simulated data to support project planning and risk mitigation [1–15].

2. Theoretical Framework

2.1. Objectives and Principles

Lifecycle-integrated governance: use ISO/IEC/IEEE 15288 as the backbone; align inputs/outputs, responsibility, and metrics from concept through disposal; break phase walls via continuous feedback [2][4].

Model-driven and data continuity: treat MBSE artifacts-requirements, functions, physical architecture, interfaces, and verification-as the authoritative source; enforce unique IDs and versions; connect tools via APIs or an event bus to form the digital thread from requirements to in-service telemetry [1][3][5].

Metric closure and risk burn-down: bind KPIs and technical risk to each review gate; require trends, corrective actions, and re-measurement; visualize risk burn-down to verify effectiveness [6–9].

Security and compliance: domain-and-level access control, least visibility, auditable use, and masked, delayed synchronization at collaboration boundaries [10][12][15].

2.2. Five-Element Architecture: Process–Organization–Data–Tools–Metrics

Process: requirements–architecture–design–integration–verification–validation as the mainline; SRR/PDR/CDR/TRR as quality gates with model consistency, traceability, coverage, and risk status as required inputs [2][4].

Organization: a Technical Board for methods and standards; IPTs for daily engineering decisions; a CCB for impact analysis and approval; responsibilities are fixed via a RACI matrix [1][3][12].

Data: master data model and unified coding for requirements, ICD, CI, defects, tests, and evidence; trunk + release branching ensures reproducibility and replayable impacts [12][15].

Tools: interoperable MBSE, requirements, PLM/ALM, simulation management, and TDM; a single authoritative repository with references elsewhere to avoid multiple masters [1][3][5].

Metrics: focus on a small set of hard KPIs-requirement stability, interface change rate, configuration conformance, first-pass yield, MTBF growth, and lead time-with dashboards, thresholds, and accountability loops [6–9][12][15].

2.3. Coupling SRR/PDR/CDR/TRR, Three Baselines, and TRL/MRL

Map SRR to the functional baseline, PDR to the allocation baseline, CDR to the product baseline, and TRR to verification readiness; each gate requires target TRL/MRL and an evidence package [2][4][6-7].

At each gate, define maturity targets and evidence lists; before TRR, require key items to reach target TRL and provide test/simulation evidence with auditable trails [6-7].

2.4. Key Mechanisms

Configuration and change control (CCB): maintain a CI-level list and genealogy; any change triggers automated impact analysis across requirements, interfaces, verification cases, and work-in-process; the system produces diffs and tracebacks after approval [12][15].

Requirements traceability and interface management (ICD/DSM): connect requirements-functions-components-test cases; manage interfaces as ICD objects; use DSM matrices to locate high-coupling zones and reduce system-wide rework [1-5][12].

Reliability growth and FRACAS closure: unify issue coding and causal categorization; integrate design, process, and component failures; use growth models to assess effectiveness; no closure without verified corrective action [8-9][13-14].

3. Implementation Plan

3.1. Top-Level Design and Governance

Roles and responsibilities: the Technical Board sets methods, standards, and gate criteria; IPTs drive delivery and trade-offs; the CCB decides based on data; Quality/Reliability maintains metric definitions and FRACAS integrity; Security/Legal defines access policy and collaboration boundaries [1-4][10][12][15].

Table 1. Sample RACI matrix

Activity	Technical Board	IPT	CCB	Quality/Reliability	Security/Legal
Standards maintenance	R	A	C	C	C
Requirements baseline freeze	C	A	C	C	C
Interface & configuration change	C	C	A	C	C
Gate criteria maintenance	R	A	C	C	C
FRACAS closure audit	C	C	C	A	C

Institutionalization: publish Specifications for Technical Reviews & Evidence Packages, Configuration & Change Control Procedure, and FRACAS Coding & Traceability. Specify inputs-activities-outputs-metrics-audit and bind gate thresholds and nonconformance handling [2][4][12][15].

Gate inputs/outputs and criteria: SRR requires requirement models and verifiability analysis; PDR requires architecture and interface consistency; CDR requires product baseline and coverage; TRR requires test readiness and residual risk, aligned with TRL/MRL targets [4][6-7].

3.2. Digital Toolchain and Data Governance

MBSE × requirements integration: one-to-one mapping between requirement objects and model elements; prohibit shadow requirements; change requests auto-propagate through the traceability chain [1][5].

PLM/ALM/simulation/TDM integration: CI-centric view binding drawings, BOM, code, models, prototypes, and test data; APIs/event bus trigger change→build→simulate→test→archive pipelines [1][3][12].

Unified coding and versioning: a shared namespace for requirements, interfaces, CI, issues, and tests; trunk + release branches; immutable history; partner domains use mapped IDs [12][15].

Digital thread and auditability: any review decision must replay to concrete model elements and data evidence; the audit system records who decided what, when, and based on which data [3][10][12][15].

3.3. Pilot Subsystems and Rollout

Phased pilots: (1) requirements and interface management; (2) configuration/change control; (3) quality and reliability with FRACAS; (4) risk management. Each phase runs metrics and gates with an iterate–measure–correct cadence.

KPI and continuous improvement: weekly dashboards, monthly reviews, and pre-gate snapshots as admission evidence [6–9].

Security and external collaboration boundaries: three-layer architecture for classified, collaboration, and open domains; delayed synchronization with masked data; partners cannot access master data, only mapped IDs [10][12][15].

4. Expected Conclusion

4.1. Expected Effects and Improvement Paths (Illustrative)

The following figures use illustrative simulated data to show KPI definitions, trends, and dashboard styles. Replace with measured project data in practice.

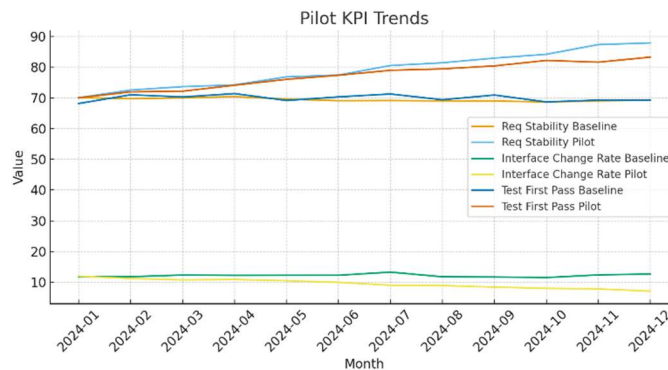


Figure 1. Pilot KPI Trends

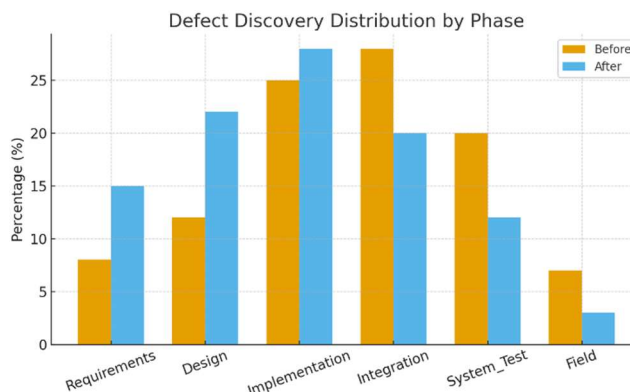


Figure 2. Defect Discovery Distribution by Phase

12-month target ranges: +10–20% requirement stability; –30–50% interface change rate; +5–10% configuration conformance; +10–15% first-pass yield; +30–60% MTBF; –10–20% lead time.

4.2. Generalizability and Limitations

Generalizability: the five-element architecture and mechanisms transfer across programs; gate-maturity mapping yields a reusable measurement-evidence-audit template.

Boundaries: depends on master data quality, toolchain integration, and personnel capability; dispersed supply chains or strict isolation require asynchronous reconciliation and delayed threading.

Risks: inconsistent metric definitions, insufficient interface automation, unclear RACI, and training gaps; mitigate via policy-first approach, structured training, staged audits, and hard thresholds.

4.3. Future Work

Empirical evaluation: A/B or regression-discontinuity across similar projects with statistical significance and effect sizes.

In-service data and digital twins: incorporate telemetry and maintenance events to close the loop from design to re-design.

AI enablement: LLM/NLP for requirement quality checks, defect prediction, and decision support as augmentation, not replacement.

References

- [1] INCOSE. Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities [M]. 5th ed. Hoboken: Wiley, 2023.
- [2] ISO/IEC/IEEE 15288:2015. Systems and software engineering-System life cycle processes [S]. Geneva: ISO, 2015.
- [3] Office of the Secretary of Defense. DoD Digital Engineering Strategy [R]. Washington D.C.: U.S. DoD, 2018.
- [4] NASA. NASA Systems Engineering Handbook [M]. Washington D.C.: NASA/SP-2016-6105 Rev2, 2016.
- [5] OMG. SysML v1.6 Specification [S]. Needham: Object Management Group, 2019.
- [6] GAO. Technology Readiness Assessment Guide [R]. GAO-20-48G, 2020.
- [7] DoD. Manufacturing Readiness Level (MRL) Deskbook [R]. Washington D.C.: OSD, 2017.
- [8] GEIA. GEIA-STD-0009: Reliability Program Standard for Systems Design, Development, and Manufacturing [S]. Arlington: SAE/GEIA, 2008.
- [9] IEC 61508:2010. Functional safety of E/E/PE safety-related systems [S]. Geneva: IEC, 2010.
- [10] GJB 9001C-2017. Quality management systems for equipment – Requirements [S]. Beijing: State Administration for Science, Technology and Industry for National Defense, 2017.
- [11] GJB 5000A-2008. Military software capability maturity model [S]. Beijing: COSTIND, 2008.
- [12] MIL-HDBK-61A. Configuration Management Guidance [S]. Washington D.C.: DoD, 2001.
- [13] IEC 60300-3-5:2001. Reliability management-Guide to FMECA [S]. Geneva: IEC, 2001.
- [14] SAE JA1000/1:2014. Reliability Program Standard Implementation Guide [S]. Warrendale: SAE, 2014.
- [15] ISO 10007:2017. Quality management-Guidelines for configuration management [S]. Geneva: ISO, 2017.

Appendix A

Table 2. KPI Definitions and Formulas (Illustrative)

Metric	Definition	Formula
Requirement Stability	Reverse measure of post-freeze changes per period	$1 - (\text{changes per period} / \text{total requirements})$
Interface Change Rate	Change requests per period divided by total managed interfaces	$\text{change requests} / \text{total interfaces}$
Configuration Conformance	Pass rate of CI checks against the baseline	$\text{passes} / \text{samples}$
First-Pass Yield	Share of test cases passing on first execution	$\text{first-pass cases} / \text{executed cases}$
Reliability Growth	MTBF improvement over time	$(\text{MTBF}_t - \text{MTBF}_0) / \text{MTBF}_0$
Lead Time	Average milestone delivery cycle	$\text{measured delivery cycle}$