Building Reliable Systems in Practice via the Integration of Model-Based and Knowledge-Centric Systems Engineering

Yousaf El Idrissi[1]
[1] Rabat School of Intelligent Computing, MOROCCO

Keywords

Building Reliable Systems
Knowledge-Centric Systems
AI

ABSTRACT

The increasing complexity and interconnectivity of modern software-intensive systems have significantly intensified the need for dependable engineering methodologies capable of ensuring reliability throughout the system lifecycle. Systems deployed in safety-critical and mission-critical domains, including aerospace, automotive, healthcare, transportation, and industrial automation, require rigorous engineering approaches to guarantee operational reliability, safety, and maintainability. Traditional engineering practices often struggle to manage the growing complexity of these systems, particularly when reliability considerations must be integrated with architectural design, requirements management, traceability, and quality assurance activities. In response to these challenges, this study proposes a comprehensive methodology that integrates Model-Based Systems Engineering (MBSE) and Knowledge-Centric Systems Engineering (KCSE) to support the practical development of dependable systems.

Introduction

Software-intensive systems have become fundamental components of modern society. Cyber-physical systems, embedded platforms, autonomous systems, intelligent transportation networks, and industrial automation infrastructures increasingly control critical operations in domains where reliability and safety are essential. The dependence of contemporary industries on interconnected software-driven systems has created substantial engineering challenges related to correctness, resilience, and operational assurance. Failures in such systems may lead not only to economic losses but also to severe safety consequences and large-scale disruptions.

The significance of dependable systems engineering is evident in many industrial incidents where software or system failures resulted in catastrophic outcomes. In aerospace, automotive, railway, and healthcare domains, failures in software-intensive systems can compromise human safety and critical infrastructure. Consequently, engineering standards such as DO-178C, ISO 26262, and EN 50128 mandate rigorous verification, validation, and reliability assurance processes throughout the system lifecycle. Reliability is therefore no longer considered a secondary quality attribute but rather a primary engineering objective that must be addressed from the earliest stages of system development.

Reliability refers to the capability of a system or component to perform its intended function under specified operating conditions for a defined period of time without failure. Achieving reliability requires systematic consideration of system architecture, environmental constraints, functional behavior, component interactions, and operational scenarios. Reliability engineering activities must therefore be integrated into broader systems engineering processes rather than being treated as isolated analyses conducted after system design completion.

Model-Based Systems Engineering (MBSE) has emerged as a prominent methodology for addressing the increasing complexity of modern systems. MBSE replaces document-centric engineering practices with formalized system models that describe system structure, functionality, interfaces, and behaviors. These models facilitate system specification, communication, analysis, and lifecycle management. Arcadia and its associated implementation platform Capella have become widely adopted MBSE solutions in industrial sectors such as aerospace, railway, defense, and automotive engineering.

Although MBSE provides strong support for architectural modeling and traceability, it often lacks advanced semantic reasoning capabilities required for managing heterogeneous engineering knowledge and reliability-related information. Knowledge-Centric Systems Engineering (KCSE) addresses this limitation by incorporating knowledge representation, ontologies, inference mechanisms, and artificial intelligence techniques into engineering activities. KCSE enables the formalization and reuse of domain knowledge while supporting semantic reasoning, automated analysis, and consistency verification.

This study proposes a methodology that integrates MBSE and KCSE to support the development of dependable systems in practice. The methodology combines Arcadia/Capella with SES Engineering Studio, a KCSE framework that supports ontology management, traceability analysis, requirements engineering, and quality assessment. The integrated methodology establishes a unified engineering environment capable of linking system models, semantic knowledge, structured requirements, and reliability-related analyses.

The primary objective of this work is to provide a practical and validated framework for reliability-focused systems engineering that leverages both model-based and knowledge-centric paradigms. The proposed methodology supports system modeling, ontology creation, structured textual requirements specification, traceability management, and model quality assessment while incorporating reliability and environmental attributes into system architectures.

Background and Related Work

Model-Based Systems Engineering and Arcadia/Capella

Model-Based Systems Engineering has become increasingly important for managing the complexity of software-intensive and cyber-physical systems. MBSE employs formalized models to represent system architecture, functionality, interfaces, and operational behavior throughout the engineering lifecycle. Unlike traditional document-centric engineering approaches, MBSE facilitates consistency, traceability, and collaborative analysis through structured system representations.

Arcadia is a widely adopted MBSE methodology developed for systems, software, and hardware architecture engineering. The methodology divides engineering activities into multiple abstraction levels, including Operational Analysis, System Analysis, Logical Architecture, Physical Architecture, and End Product Breakdown Structure. Each level provides specialized diagrams and modeling perspectives that support system decomposition and refinement.

Capella is the implementation platform for Arcadia and provides extensive support for architectural modeling. Capella includes modeling accelerators, diagram editors, and engineering workflows that enable systems engineers to focus on design activities while maintaining model consistency. Capella supports numerous diagram types, including operational capability diagrams, functional dataflow diagrams, architecture diagrams, and exchange scenario diagrams.

The flexibility of Arcadia/Capella allows organizations to customize viewpoints and incorporate additional engineering information into models. Through Capella Studio, engineers can create domain-specific viewpoints that extend standard model elements with additional attributes and properties relevant to reliability, safety, security, or environmental analysis.

Knowledge-Centric Systems Engineering and SES Engineering Studio

Knowledge-Centric Systems Engineering extends traditional systems engineering by incorporating semantic technologies, ontologies, and artificial intelligence methods into engineering processes. KCSE recognizes that engineering activities depend heavily on domain knowledge, terminology, relationships, and reasoning mechanisms that must be explicitly represented and managed.

SES Engineering Studio is a commercial KCSE platform designed to support interoperability among diverse engineering tools and processes. SES integrates ontology management, requirements engineering, traceability analysis, verification and validation management, and quality assessment capabilities within a unified environment.

The SES framework includes several specialized modules. RQA Quality Studio supports artifact quality analysis and reporting. Verification and Validation Studio manages assurance activities and validation workflows. Traceability Studio supports trace management and automatic identification of missing traces. Knowledge Manager enables ontology creation and maintenance. Requirements Authoring Tools facilitate structured textual requirements engineering.

Ontologies within SES are composed of terminology definitions, conceptual models, patterns, formal semantic representations, and inference rules. These components enable automated reasoning, semantic consistency checking, and advanced traceability analysis across engineering artifacts.

Related Research

Previous research has explored various aspects of reliability engineering, MBSE, ontology-driven engineering, and tool integration. Studies on model-based safety analysis, risk management, verification, and traceability have demonstrated the value of formal system models for engineering assurance activities. Similarly, ontology-based approaches have been investigated for requirements engineering, software reliability analysis, and semantic traceability management. Several studies have attempted to integrate models and ontologies within systems engineering environments. These efforts include ontology-driven traceability frameworks, semantic model management techniques, and knowledge-enhanced verification systems. However, most existing studies focus either on generic systems engineering applications or on isolated reliability analyses rather than on comprehensive methodologies for dependable system design.

Research involving Arcadia/Capella has addressed safety analysis, cybersecurity assessment, verification support, and architectural assurance. Likewise, SES has been used for requirements quality analysis, semantic traceability, and standards compliance management. Nevertheless, the integration of Arcadia/Capella and SES specifically for reliability-focused engineering has received limited attention.

This study extends previous work by proposing and validating a unified methodology that integrates MBSE and KCSE to support dependable systems engineering in industrial practice.

Proposed Methodology

The proposed methodology integrates Arcadia/Capella with SES Engineering Studio to establish a unified framework for reliability-focused systems engineering. The methodology consists of five major engineering activities: system modeling, ontology creation, structured textual requirements specification, traceability management, and model quality assessment.

System Modeling

The first activity involves creating system models within Arcadia/Capella. Depending on project objectives, engineers may employ operational, system, logical, or physical architecture diagrams to represent system structure and functionality.

A major contribution of the methodology is the extension of Capella through reliability-oriented viewpoints. These viewpoints allow engineers to associate reliability and environmental attributes directly with model elements. Reliability attributes include mean time between failures, failure rates, and operational reliability parameters. Environmental attributes include temperature, humidity, voltage constraints, and operating conditions.

Functional diagrams additionally incorporate response-time characteristics and performance-related attributes. These extensions support early reliability analysis and ensure that reliability considerations remain integrated with system architecture throughout the development lifecycle.

Ontology Creation

Ontology creation is performed within SES Engineering Studio. Engineers define terminology, semantic relationships, conceptual structures, and inference rules associated with the target domain. The ontology captures knowledge related to system components, reliability characteristics, environmental conditions, operational constraints, and engineering standards. This semantic representation supports automated reasoning, consistency checking, and advanced traceability analysis.

Structured Requirements Specification

The methodology supports structured textual requirements specification using SES authoring tools. Requirements are formalized according to predefined semantic patterns that improve consistency, clarity, and traceability.

Structured requirements facilitate automated analysis and reduce ambiguity in system specifications. The approach also supports compliance with industrial standards requiring rigorous requirements engineering practices.

Traceability Management

Traceability management connects system models, requirements, ontological knowledge, and reliability attributes. SES Traceability Studio enables engineers to establish and maintain semantic links among engineering artifacts.

Automated trace analysis identifies missing relationships, inconsistencies, and potential gaps in verification coverage. Improved traceability enhances reliability assurance and supports certification activities.

Model Quality Assessment

The methodology incorporates quality assessment activities using SES quality analysis capabilities. Quality metrics evaluate model completeness, consistency, traceability, and compliance with engineering standards.

The combination of MBSE and KCSE enables comprehensive analysis of both structural model quality and semantic consistency. This integrated assessment improves engineering confidence and supports reliability-oriented decision-making.

Validation and Evaluation

The methodology was validated through application to eight systems across five industrial domains. Validation activities included architectural modeling, ontology development, traceability management, and reliability-focused quality assessment.

The evaluation demonstrated that the methodology can support diverse engineering scenarios while maintaining strong reliability assurance capabilities. Engineers successfully integrated reliability attributes into architectural diagrams and established semantic relationships between requirements, system models, and ontological knowledge.

Validation results further indicated that the methodology improved traceability management and facilitated more systematic reliability analysis. The integrated environment supported compliance-oriented engineering workflows consistent with safety-critical standards such as DO-178C, ISO 26262, and EN 50128.

The results additionally highlighted the benefits of combining structural modeling with semantic reasoning. Engineers were able to identify inconsistencies, missing traces, and quality issues more effectively than with isolated MBSE or documentation-based approaches.

Discussion

The study demonstrates that integrating MBSE and KCSE provides substantial benefits for dependable systems engineering. Arcadia/Capella offers strong architectural modeling capabilities, while SES Engineering Studio contributes semantic reasoning, ontology management, and AI-supported quality assessment.

The methodology addresses important industrial challenges associated with managing reliability-related knowledge, maintaining traceability, and supporting compliance activities. By embedding reliability information directly into system models and connecting those models with semantic knowledge bases, the approach improves engineering consistency and lifecycle management.

The integration of AI-driven semantic technologies further enhances the ability to manage complex engineering knowledge and perform automated reasoning tasks. This is particularly important for large-scale cyber-physical and software-intensive systems characterized by high complexity and extensive interdependencies.

Despite these benefits, several challenges remain. The methodology requires specialized expertise in both MBSE and ontology engineering. Scalability considerations may also arise for extremely large system architectures and extensive semantic knowledge bases. Furthermore, advanced quantitative reliability analyses still depend on external analysis tools.

Conclusion

This study presented a comprehensive methodology for developing dependable systems through the integration of Model-Based Systems Engineering and Knowledge-Centric Systems Engineering. By combining Arcadia/Capella with SES Engineering Studio, the methodology establishes a unified engineering framework that supports architectural modeling, ontology management, structured requirements engineering, traceability analysis, and quality assessment. The proposed approach demonstrates that reliability engineering can be significantly enhanced through the integration of formal system models and semantic knowledge representation. Reliability attributes and environmental constraints can be incorporated directly into architectural diagrams, while ontology-driven reasoning mechanisms improve consistency, traceability, and automated quality analysis. Validation across multiple industrial domains confirmed the practicality and flexibility of the methodology. The results showed that the integrated MBSE-KCSE framework supports reliability-focused engineering activities while facilitating compliance with safety-critical standards and improving engineering traceability.

Future research may focus on expanding automation capabilities, improving scalability, integrating quantitative reliability analysis tools, and supporting emerging domains such as autonomous systems, digital twins, and cloud-native cyber-physical infrastructures. The integration of advanced machine learning techniques may further enhance semantic reasoning, anomaly detection, and automated engineering support.

Overall, the study demonstrates that the integration of MBSE and KCSE provides a viable and effective foundation for designing dependable software-intensive and cyber-physical systems in modern industrial environments.

References

[1] Lopez, B., Alvarez-Rodriguez, J. M., Parra, E., & de la Vara, J. L. (2020, June). Ontology configuration management for knowledge-centric systems engineering in industry. In *2020 50th Annual IEEE-IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S)* (pp. 37-40). IEEE.

[2] Kuntamukkala, N. K., & Thalary, S. (2021). Self-Optimizing Angular Applications: A Novel Framework for AI-Driven Performance Adaptation in Production Environments. *International Journal of AI, BigData, Computational and Management Studies*, *2*(2), 107-117.

[3] Wang, R., Milisavljevic-Syed, J., Guo, L., Huang, Y., & Wang, G. (2021). Knowledge-based design guidance system for cloud-based decision support in the design of complex engineered systems. *Journal of Mechanical Design*, *143*(7), 072001.

[4] Thalary, S., & Katipelly, A. (2021). CI/CD for Distributed Software Systems: Why Software Architecture Determines Pipeline Complexity. *International Journal of Emerging Research in Engineering and Technology*, *2*(4), 100-111.

[5] El Alaoui, M., Rabah, S., Chapurlat, V., Richet, V., & Plana, R. (2022). An original Data, Information and Knowledge management approach for model-based engineering projects. *IFAC-PapersOnLine*, *55*(10), 3262-3267.

[6] Thalary, S., & Kuntamukkala, N. K. (2022). Operationalizing Software Invariants: A DevOps-Driven Approach to Reliability in Cloud-Native Systems. *International Journal of Emerging Trends in Computer Science and Information Technology*, *3*(4), 157-168.

[7] Borchani, M. F., Hammadi, M., Ben Yahia, N., & Choley, J. Y. (2019). Integrating model-based system engineering with set-based concurrent engineering principles for reliability and manufacturability analysis of mechatronic products. *Concurrent Engineering*, *27*(1), 80-94.

[8] Thalary, S. (2022). Cloud Cost, Reliability, and Speed: The Triangle Every Enterprise Struggles With. *International Journal of Emerging Research in Engineering and Technology*, *3*(4), 141-152.

[9] Lee, C. H., Chen, C. H., Li, F., & Shie, A. J. (2020). Customized and knowledge-centric service design model integrating case-based reasoning and TRIZ. *Expert Systems with Applications*, *143*, 113062.

[10] Thalary, S., & Katipelly, A. (2023). Secure-by-Design Cloud Software Delivery: How DevOps and Software Teams Co-Own Security Outcomes. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *4*(1), 131-140.

[11] Mu'azzam, K., da Silva, F. V. S., Murtagh, J., & Gallagher, M. J. S. (2024). A roadmap for model-based bioprocess development. *Biotechnology Advances*, *73*, 108378.
[12] Katipelly, A., & Thalary, S. (2023). Cryptographic Identity Propagation in Asynchronous Event-Driven Architectures: Implementing Zero-Trust Envelopes for High-Velocity Payment Streams. *International Journal of Emerging Trends in Computer Science and Information Technology*, *4*(2), 212-222.
[13] Harrison, N. A. W., Whitfield, R. I., Odukoya, K. A., & Powell, A. I. (2022, November). Model-based engineering and the digital thread. In *16th International Naval Engineering Conference*.
[14] Thalary, S. (2023). Monitoring Isn't Observability: Lessons from Running Enterprise Microservices. *International Journal of Emerging Research in Engineering and Technology*, *4*(2), 139-148.
[15] Lee, J., Suh, E. H., & Hong, J. (2010). A maturity model based CoP evaluation framework: A case study of strategic CoPs in a Korean company. *Expert Systems with Applications*, *37*(3), 2670-2681.
[16] Thalary, S. (2024). From Pipelines to Policy: Embedding AI-Ready Governance into Cloud DevOps at Scale. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *5*(1), 200-210.
[17] Bögel, S., Stieglitz, S., & Meske, C. (2014). A role model-based approach for modelling collaborative processes. *Business Process Management Journal*, *20*(4), 598-614.
[18] Thalary, S., & Katipelly, A. (2024). Cloud-Native Design for Event-Driven Systems: Where Software Architecture Decisions Meet DevOps Reality. *International Journal of AI, BigData, Computational and Management Studies*, *5*(2), 202-212.
[19] Patwari, P., Choudhury, S. R., Banerjee, A., Swaminathan, N., & Pandey, S. (2016, July). M&C Domain Map Maker: an environment complimenting MDE with M&C knowledge and ensuring solution completeness. In *Software and Cyberinfrastructure for Astronomy IV* (Vol. 9913, pp. 1426-1437). SPIE.
[20] Katipelly, A., & Thalary, S. (2024). Semantic Automation of Basel III Liquidity Reporting: Utilizing Ontological Knowledge Graphs for Real-Time Regulatory Compliance and Auditability. *International Journal of Emerging Research in Engineering and Technology*, *5*(2), 147-156.
[21] Dori, D. (2013, January). Conceptual Modeling Semantics for the Physical-Informatical Essence Duality Problem. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*.
[22] Kuntamukkala, N. K., & Thalary, S. (2024). Intelligent Angular Architecture: Machine Learning-Based Component Recommendation Systems for Enterprise-Scale Development. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *5*(4), 276-284.
[23] Lenord, O. (2024). OpenSCALING-A European effort to scale-up model-based systems engineering to the needs of carbon neutral transformation. *Center for Model-Based Cyber-Physical Product Development*, *33*(18).

[24] Thalary, S., & Katipelly, A. (2025). Platform Engineering for Distributed Systems: How Cloud DevOps Enables Scalable, Policy-Driven Software Architectures. *International Journal of AI, BigData, Computational and Management Studies*, *6*(1), 189-197.

[25] Abidi, S. S. R. (2007, July). Healthcare knowledge management: The art of the possible. In *AIME workshop on knowledge management for health care procedures* (pp. 1-20). Berlin, Heidelberg: Springer Berlin Heidelberg.

[26] Katipelly, A., & Thalary, S. (2025). Carbon-Aware Dynamic Batching for Deep Learning Inference: Optimizing the Energy-Latency Trade-off in High-Frequency Transaction Monitoring. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *6*(3), 160-169.

[27] Natarajan, S., Chaudhuri, S. R., & Kumar, A. (2019, December). Modular semantically integrated system modelling. In *INCOSE International Symposium* (Vol. 29, pp. 42-59).

[28] Thalary, S. S. D., Kuntamukkala, N. K., Chennareddy, R. K., & Manoj, M. S. (2026, March). DevOps Excellence for Enterprise AI Systems: Continuous Deployment of Resilient Platforms With Angular Frontend Integration. In *2026 IEEE Madhya Pradesh Section Conference (MPCON)* (pp. 1143-1151). IEEE.

[29] Roudier, Y., Idrees, M. S., & Apvrille, L. (2014, May). Improved security requirements engineering using knowledge representation. In *9ème conférence sur la Sécurité des Architectures Réseaux et des Systèmes d'Information*.

[30] Kuntamukkala, N. K., Chennareddy, R. K., Thalary, S. S. D., & Manoj, M. S. (2026, March). Building Production-Grade Angular Frontends for Enterprise AI: Resilient Architecture Patterns and Devops-Enabled Deployment Strategies. In *2026 IEEE Madhya Pradesh Section Conference (MPCON)* (pp. 1099-1103). IEEE.

[31] Hoffmann, D., Nowacki, N., Biffl, S., Kiesling, E., Meixner, K., & Lüder, A. (2023). Interdisciplinary production risk exploration: A grounded approach to integrate data- and knowledge-driven analytics. *Procedia CIRP*, *120*, 1016-1021.

[32] Chennareddy, R. K., Thalary, S. S. D., Kuntamukkala, N. K., & Manoj, M. S. (2026, March). Designing Resilient Enterprise AI Platforms: Integrating Angular-Based Intelligent Interfaces with DevOps-Driven Operational Excellence. In *2026 IEEE Madhya Pradesh Section Conference (MPCON)* (pp. 1137-1142). IEEE.