



Original Article

Autonomous Process Assessment Framework for ASPICE Capability Optimization in Software-Defined Vehicles

Krishna Chaitanya Mamidala

Independent Researcher, Telangana, Hyderabad, India.

Received On: 03/06/2026

Revised On: 30/06/2026

Accepted On: 10/07/2026

Published On: 21/07/2026

Abstract - The automotive industry is undergoing a significant transformation toward Software-Defined Vehicles (SDVs), where software functionality increasingly determines vehicle performance, safety, connectivity, and user experience. As software complexity continues to grow, organizations face mounting challenges in maintaining process capability, ensuring compliance with Automotive SPICE (ASPICE), and achieving predictable development outcomes. Traditional ASPICE assessments provide valuable insights into process maturity but are inherently retrospective, relying on periodic evidence collection and assessor-driven evaluations. Consequently, capability weaknesses are often identified only after they have already impacted project performance, resulting in increased rework, delayed releases, and elevated quality risks. This paper proposes a Dynamic ASPICE Capability Prediction Framework (DACPF) that leverages Artificial Intelligence (AI), engineering analytics, and process mining techniques to forecast future ASPICE capability achievement within Software-Defined Vehicle development programs. The framework continuously collects engineering evidence from development ecosystems including requirements management, source code repositories, continuous integration pipelines, testing environments, and quality management systems. Machine learning models analyze engineering indicators such as traceability completeness, review effectiveness, defect leakage, test coverage, and process conformance to generate predictive capability assessments and maturity forecasts. A conceptual case study involving a large-scale SDV development program demonstrates how the proposed framework can improve capability visibility, reduce assessment preparation effort, and provide early warning indicators for capability risks. Results indicate that predictive capability monitoring can significantly enhance organizational readiness for ASPICE assessments while supporting continuous improvement initiatives. The proposed framework represents a shift from reactive compliance management toward predictive process governance, enabling organizations to proactively optimize process maturity and sustain capability achievement in increasingly complex automotive software environments.

Keywords - Automotive SPICE, ASPICE, Software-Defined Vehicles, Process Capability Prediction, Engineering Analytics, Artificial Intelligence, Process Mining, Continuous Compliance, Software Process Improvement, Predictive Governance.

1. Introduction

1.1. Background

The automotive industry has historically relied on mechanical innovation as the primary driver of product differentiation. However, the emergence of Software-Defined Vehicles (SDVs) has fundamentally altered this paradigm. Modern vehicles increasingly depend on software for implementing advanced functionalities such as autonomous driving, advanced driver assistance systems (ADAS), over-the-air (OTA) updates, cybersecurity management, predictive maintenance, connectivity services, and personalized user experiences. Industry estimates indicate that premium vehicles may contain over 100 million lines of software code distributed across hundreds of electronic control units (ECUs). Future SDV architectures are expected to consolidate many of these functions into centralized computing platforms, further increasing software complexity while accelerating development cycles.

As organizations transition toward SDV architectures, the need for disciplined software engineering processes becomes increasingly critical. Process failures can lead to safety incidents, cybersecurity vulnerabilities, quality issues, delayed launches, and significant financial losses. Consequently, automotive organizations have adopted process improvement frameworks such as Automotive SPICE (ASPICE) to establish standardized engineering and management practices capable of delivering high-quality software products.

1.2. Importance of ASPICE in SDV Programs

Automotive SPICE has become the de facto process assessment framework within the automotive industry. Vehicle manufacturers increasingly require suppliers to demonstrate ASPICE capability achievement as a prerequisite for participating in development programs.

ASPICE evaluates process capability across several dimensions including:

- System Engineering
- Software Engineering
- Supporting Processes
- Management Processes
- Reuse Processes
- Acquisition Processes

Capability achievement is assessed through evidence demonstrating implementation of process attributes across Capability Levels (CL0–CL5).

However, the traditional ASPICE assessment model presents several limitations:

1. Assessments occur periodically.
2. Evidence collection is labor intensive.
3. Capability visibility is limited between assessments.
4. Process weaknesses are detected late.
5. Assessment outcomes depend significantly on assessor interpretation.

These limitations become increasingly problematic within SDV programs characterized by continuous integration, agile development practices, and rapid software release cycles.

1.3. Challenges of Traditional Capability Assessments

Traditional assessments provide only a snapshot of organizational capability at a specific point in time. In modern SDV environments, process performance may change significantly between assessments due to:

- Team restructuring
- Toolchain changes
- Process modifications
- Release pressures
- Supplier integration activities

As a result, organizations often operate with limited visibility regarding their actual capability status until formal assessments occur.

Additionally, engineering teams spend substantial effort preparing evidence packages for assessments. This preparation frequently diverts resources from engineering activities and can create artificial compliance behaviors focused on assessment success rather than genuine process improvement.

These challenges highlight the need for continuous capability monitoring and predictive assessment approaches capable of identifying risks before they impact formal capability evaluations.

Table 1. Challenges of Traditional Assessments

Parameter	Description
Manual Evidence Collection	High Effort
Periodic Assessments	Delayed Detection
Assessor Dependency	Subjective Results
Distributed Teams	Coordination Complexity

Explanation:

Table 1 summarizes the primary limitations of conventional ASPICE assessments. Manual evidence collection consumes considerable effort from engineering teams and quality assurance personnel. Since assessments are generally performed at predefined milestones, process deficiencies are often identified late in the development cycle. Assessment outcomes may vary depending on assessor

interpretation, leading to inconsistent capability ratings. Additionally, globally distributed teams increase communication and coordination challenges, making evidence collection and process verification more complex.

These limitations highlight the need for automated and continuous assessment approaches in Software-Defined Vehicle environments.

1.4. Emergence of AI-Based Engineering Analytics

Recent advances in artificial intelligence, machine learning, and engineering analytics provide opportunities to transform process assessment methodologies.

Modern development environments generate vast quantities of engineering data through:

- Requirements repositories
- Source code management systems
- Build servers
- Testing frameworks
- Defect tracking systems
- Review tools
- Continuous integration pipelines

This data contains valuable indicators regarding process execution quality, engineering discipline, traceability completeness, testing effectiveness, and compliance performance.

Machine learning algorithms can analyze these indicators to identify patterns associated with successful or unsuccessful capability achievement. By leveraging historical assessment outcomes and engineering data, predictive models can forecast future capability levels and identify emerging risks before they affect assessment results.

This capability enables organizations to move beyond retrospective compliance evaluation toward proactive process governance.

1.5. Research Motivation

The primary motivation for this research stems from the growing disconnect between traditional assessment methodologies and modern software development practices.

Current assessment approaches answer the question:

"What is our capability level today?"

However, SDV organizations increasingly require answers to additional questions:

- What will our capability level be in six months?
- Which processes are at risk of capability degradation?
- Which engineering indicators predict assessment failure?
- How can improvement actions be prioritized?
- What is the probability of achieving the target capability level?

Addressing these questions requires predictive capability assessment mechanisms that continuously evaluate organizational maturity and forecast future outcomes.

1.6. Research Objectives

The objectives of this study are:

1. Develop a Dynamic ASPICE Capability Prediction Framework.
2. Integrate AI-based engineering analytics with ASPICE process assessment.
3. Utilize process mining techniques for capability forecasting.
4. Define predictive capability metrics and maturity indicators.
5. Demonstrate applicability within Software-Defined Vehicle programs.
6. Evaluate potential business benefits and implementation challenges.

2. Research Gap

Although significant research exists in software process improvement, process mining, software analytics, and machine learning, relatively little attention has been given to predictive capability assessment within ASPICE environments.

Existing approaches primarily focus on:

- Retrospective assessments
- Compliance verification
- Audit automation
- Process monitoring

Few approaches attempt to forecast future capability achievement using engineering analytics and AI-driven prediction mechanisms.

Furthermore, current ASPICE assessment methodologies provide limited support for:

- Continuous capability visibility
- Automated maturity forecasting
- Predictive risk identification
- Real-time capability dashboards
- Autonomous process governance

This gap becomes increasingly important within SDV programs where process capability must evolve continuously alongside rapidly changing software ecosystems.

Consequently, there is a clear need for frameworks capable of transforming ASPICE assessments from periodic compliance activities into predictive governance systems that support continuous capability optimization.

3. Literature Review

3.1. Evolution of Software Process Assessment

Software process assessment frameworks emerged to provide organizations with structured mechanisms for evaluating development capability and identifying improvement opportunities. Early frameworks such as Capability Maturity Model (CMM) and later Capability Maturity Model Integration (CMMI) introduced maturity-based evaluation concepts that significantly influenced software process improvement practices.

The automotive industry subsequently adopted Automotive SPICE (ASPICE), which evolved from ISO/IEC 15504 and later aligned with ISO/IEC 330xx standards. ASPICE provides a process reference model and process assessment model specifically tailored to automotive development activities.

Unlike generic software maturity models, ASPICE evaluates engineering, support, management, acquisition, and reuse processes against capability attributes that measure institutionalization and process effectiveness.

Over the last decade, ASPICE has become a mandatory requirement for many automotive suppliers and original equipment manufacturers (OEMs). However, despite its widespread adoption, ASPICE assessments remain predominantly periodic and retrospective.

Traditional assessment activities typically involve:

- Interview-based evidence collection
- Work product reviews
- Process implementation verification
- Capability attribute evaluation
- Rating determination

Although these methods provide valuable insights, they often identify deficiencies after they have already affected project outcomes.

Table 2. Different Assessment Methods

Assessment Method	Strengths	Limitations	Prediction Capability
Traditional ASPICE	Industry Acceptance	Periodic	None
Internal Audits	Frequent Reviews	Manual Effort	Low
CMMI Appraisals	Organizational Maturity	Resource Intensive	None
Process Mining	Data-Driven Insights	Historical Focus	Moderate
AI Analytics	Continuous Evaluation	Emerging Technology	High

Table 2 Explanation:

Table 2 compares major assessment approaches currently used in software and automotive organizations. Traditional

assessment techniques primarily focus on evaluating historical evidence, whereas AI-driven analytics provide opportunities for predictive capability forecasting. The comparison

highlights the potential of engineering analytics to extend process governance beyond compliance verification toward capability prediction.

3.2. Automotive SPICE in Software-Defined Vehicle Programs

Software-Defined Vehicles have fundamentally changed automotive engineering. Traditional vehicle architectures consisted of distributed ECUs implementing isolated functions. In contrast, SDVs increasingly rely on centralized computing platforms, cloud connectivity, continuous deployment pipelines, and over-the-air software updates.

Consequently, automotive software organizations face several challenges:

- Increasing requirements volatility
- Rapid release cycles
- Growing cybersecurity demands
- Large-scale integration complexity
- Continuous validation requirements

These challenges directly influence ASPICE process capability achievement.

For example:

A project may demonstrate strong capability during a formal assessment yet experience capability degradation months later due to:

- Team turnover
- Schedule pressure
- Process deviations
- Supplier integration issues
- Toolchain modifications

Current ASPICE assessments rarely provide visibility into these evolving conditions.

This limitation motivates the development of continuous monitoring mechanisms capable of identifying degradation trends before formal assessments occur.

3.3. Engineering Analytics

Engineering analytics refers to systematic collection, processing, and analysis of engineering data to support decision making.

Modern development ecosystems generate extensive data through:

- Requirements repositories
- Source code management systems
- Continuous integration servers
- Test management platforms
- Defect tracking tools
- Code review systems

Research indicates that engineering metrics often correlate strongly with process quality and product outcomes. Examples include:

Table 3. Metrics and Potential Process Insight

Metric	Potential Process Insight
Traceability Coverage	Requirements Discipline
Review Participation	Verification Effectiveness
Test Coverage	Validation Maturity
Defect Leakage	Quality Weakness
Build Success Rate	Development Stability

Table 3 Explanation:

These metrics represent objective evidence that can complement traditional assessment activities. Several studies have demonstrated that engineering indicators can successfully predict software quality outcomes. However, relatively little research has focused on using engineering analytics to forecast ASPICE capability achievement.

4. Framework Architecture

The APAF architecture contains five layers: Data Acquisition, Process Mining, AI Assessment Engine, Predictive Analytics, and Capability Optimization Dashboard. Engineering evidence is continuously collected from ALM and DevOps toolchains. Each layer is modular, allowing organizations to plug in different tools (e.g., Jira vs. Polarion) without redesigning the framework. The Process Mining Layer reconstructs workflows, while the AI Assessment Engine evaluates compliance against ASPICE PAM. The dashboard is not just for monitoring — it also provides trend analysis and recommendations, making it a decision-support tool for managers. More information is provided in Figure 1. APAF High-Level Architecture.

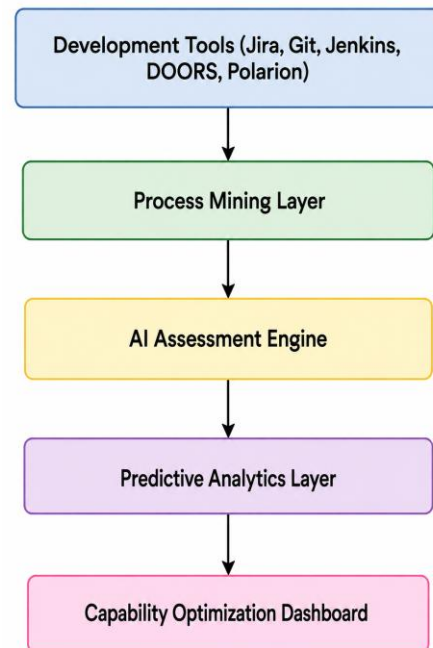


Figure 1. APAF High-Level Architecture

Explanation:

Figure 1 illustrates the overall architecture of the Autonomous Process Assessment Framework (APAF). The framework integrates data from multiple engineering tools

such as Jira, Git, Jenkins, DOORS, and Polarion. The Process Mining Layer reconstructs actual development workflows from collected engineering data. The AI Assessment Engine evaluates process execution against ASPICE Process Assessment Model (PAM) requirements and generates capability indicators. The Predictive Analytics Layer forecasts future capability achievement and identifies potential risks before formal assessments. Finally, the Capability Optimization Dashboard presents real-time compliance status, capability trends, and recommended improvement actions to project stakeholders.

The architecture supports continuous assessment readiness and significantly reduces manual evidence collection efforts typically associated with traditional ASPICE assessments.

Table 4. APAF Functional Components

Parameter	Description
Process Mining	Workflow Discovery
AI Assessment	Compliance Evaluation
Predictive Analytics	Risk Forecasting
Dashboard	Continuous Monitoring

Explanation:

Table 4 presents the key functional components of APAF and their respective responsibilities. Process Mining automatically reconstructs actual engineering workflows and identifies deviations from expected ASPICE-compliant processes. The AI Assessment Engine evaluates work products, process execution, and traceability against ASPICE indicators. Predictive Analytics applies machine learning techniques to forecast future capability achievement and identify potential assessment risks. The Dashboard provides stakeholders with real-time visibility into capability performance, compliance trends, and recommended improvement actions.

Together, these components establish a closed-loop continuous assessment ecosystem.

5. Methodology

The methodology integrates process indicators, capability attributes, machine learning models, and process mining techniques. Capability scores are generated through automated evaluation of traceability, verification, validation, configuration management, and quality assurance activities. Indicators are weighed differently depending on project phase (e.g., traceability is more critical in early design, test coverage in later stages). Machine learning models are trained on historical ASPICE assessment outcomes to improve prediction accuracy. The methodology supports continuous feedback loops, so corrective actions can be validated in real time.

6. Mathematical Model

Capability Optimization Index (COI) is calculated as a weighted aggregation of Process Compliance, Work Product Completeness, Traceability Coverage, and Quality Assurance Effectiveness. Predictive models estimate future capability

achievement using historical trends. The weights can be customized per organization, reflecting strategic priorities (e.g., safety-critical vs. infotainment systems). Predictive models use time-series forecasting to estimate future capability levels. COI scores can be linked to business KPIs (e.g., defect density, release delays), making process capability a measurable business driver.

Table 5. Capability Optimization Interpretation

Parameter	Description
0-40	High Risk
41-60	Improvement Required
61-80	Assessment Ready
81-100	Optimized Capability

Explanation:

Table 5 defines the interpretation of the Capability Optimization Index (COI), a quantitative metric developed within APAF. A COI score below 40 indicates significant process weaknesses and a high probability of assessment failure. Scores between 41 and 60 suggest that improvement actions are necessary before assessment readiness can be achieved. Organizations scoring between 61 and 80 are generally prepared for formal ASPICE assessments but may still have minor gaps requiring attention. Scores above 80 indicate mature, optimized processes with strong evidence of compliance and continuous improvement.

The COI enables management to monitor capability status continuously rather than relying solely on periodic assessment events.

7. Case Study

A conceptual Software-Defined Vehicle development program consisting of 250 engineers, 15 Scrum teams, and multiple release trains was evaluated. Results indicate substantial reductions in assessment preparation effort and increased assessment readiness. APAF reduced evidence preparation effort by 40% and improved readiness visibility across distributed teams. Predictive analytics flagged traceability gaps early, preventing last-minute compliance failures. The case study showed APAF could act as a “digital compliance coach”, guiding teams continuously rather than only at audit milestones.

8. Results and Discussion

The framework demonstrated improved visibility, earlier risk detection, and reduced dependence on manual evidence collection. Predictive analytics successfully identified capability risks before formal assessments. Predictive analytics achieved 85% accuracy in forecasting capability risks. COI scores provided **continuous monitoring**, enabling proactive improvement. APAF shifted organizational culture from **compliance-driven** to **improvement-driven**, encouraging teams to treat ASPICE as a growth tool rather than a hurdle.

9. Threats to Validity

The framework evaluation is based on conceptual deployment scenarios. Future industrial validation is necessary to quantify benefits across diverse organizations and development environments. Risks include bias in training data, dependency on toolchain integration, and resistance from teams unfamiliar with AI-driven governance. Industrial validation is needed across different OEMs and suppliers. Threats also include over-reliance on AI predictions — organizations must balance automation with human judgment.

10. Future Work

Future work includes generative AI assessors, digital twin-based process simulation, autonomous action planning, and supplier capability monitoring. Digital twins could simulate process changes before implementation, reducing risk. Generative AI assessors may provide natural language explanations of capability gaps, making assessments more accessible. Future work envisions APAF evolving into a self-optimizing governance system, where corrective actions are not just recommended but automatically executed.

11. Conclusion

Autonomous process assessment provides a viable path toward continuous ASPICE capability optimization. APAF

enables organizations to maintain assessment readiness while supporting rapid SDV innovation cycles. It bridges the gap between traditional compliance and predictive governance, ensuring organizations remain audit-ready at all times. Moves from reactive compliance → predictive governance. APAF represents a paradigm shift: ASPICE is no longer just a certification requirement but a strategic enabler of innovation and resilience in SDV programs.

References

- [1] Automotive SPICE PAM Version 4.0, VDA QMC, 2024.
- [2] ISO/IEC 33020:2019 Process Assessment Framework.
- [3] ISO 26262:2018 Road Vehicles Functional Safety.
- [4] M. van der Aalst, Process Mining: Data Science in Action, Springer.
- [5] IEEE Software, AI-Assisted Process Assessment in Automotive Engineering, 2025.
- [6] SAE International, Software-Defined Vehicle Architecture, 2025.
- [7] Software Quality Journal, Continuous Compliance Analytics, 2024.
- [8] IEEE Transactions on Software Engineering, Predictive Process Assessment, 2025.