Membrane Technology

ISSN (online): 1873-4049

Software-Defined Vehicles: Transforming Automotive Engineering Through Modular Architecture And Ai-Enabled Validation

Satyabrata Pradhan

Sr. Software Engineer/Senior Program Manager

Abstract

The research of this paper aims at discussing the prospects of the change of the world of automotive engineering with the help of Software-Defined Vehicles (SDVs) with the help of modular architecture, AI-assisted validation and model-based systems engineering (MBSE). SDVs will formal detection digitalized divide by centralized, zonal plan which will reduce wiring by up to 45 per cent using 2-dose of the mean duration between failures that typical structures. It has hit speeds of 190 MB/s with Cellular Vehicle-to-Everything (C-V2X) and 5G transmission rate and thus complete software updates with OTA which is reportedly 70 times quicker than systems controlled via Wi-Fi and 5000 times more adept than software update. Comparison between AI based validation and scripted coverage gave 81-percent bad code coverage compared to 62-percent scripted coverage at half the time. The requirements traceability score has achieved 25 points out of the required 78 to 97percent and preparations in certifications have reduced by 40 percent because of MBSE integration.

This means that SDVs do not have to be commodities, but instead can be constructed and delivered as running platforms. The findings also indicate that AI and MBSE also minimize engineering labour and attain a safety, reliability, user-trust increase. Study revealed that it required incorporation of modular architecture and intelligent testing, among others, so as to keep abreast of the increasing complexity of the automotive systems. Among its implications are the decrease in the cost of maintenance, the steepening of the timeline of software adoption, the growth of the level of trust in autonomous or semi-autonomous vehicles among the population.

KEYWORDS: Validation. Software-defined Vehicles, AI, Automation

I. INTRODUCTION

The auto market is transforming the hardware-based design to the software-based design. Similar-minded, driver assistance, infotainment, connectivity and electric powertrain system software, instead of all the hardware in cars, exist. This has led to a much different concept of Software-Defined Vehicles (SDVs) where behaviour and capabilities can be enhanced with code, not hardware recalls.

The engineers, testing, and maintenance of vehicles is made even harder through the new model. Conventional systems with extensive numbers of autonomous electronic control modules (ECUs) involve high wiring overhead as well as additional malfunctions; and larger amounts of discontinuities. Meanwhile, the needs of clients in the condition of periodicity of opportunities in the perpetual development of functions and extremely high safety standards are increasing.

These problems are discussed in three critical ways (modular zonal architectures, AI assisted validation and model-based systems engineering (MBSE)) in the paper. The implication of the modular zonal systems is also of reduced wires and, by extension, simpler results of the system integration, where the results are in the form of Over-the-Air (OTA) program updates that mop down the cars already updated in the field. The validation associated with the AI-based reinforcement learning-like agent allows testing the behaviors of the vehicles and identifying the defects more effectively than any laboratory testing. The approach of MBSE connects the system requirements and continuous the integration pipelines, the testing pipelines that advances the traceability and certification preparedness.

Such methods discover a means to create an edifice to rely on and protect SDVs as movable bases. Our results of the impact of these technologies on the enhancement of performance, the decrease of the cost, and promotion of the confidence towards the new generation car models is preconditioned by this background.

II. RELATED WORKS

Software-Defined Vehicles

The notion of Software Defined Vehicles (SDVs) can be regarded as a new concept upending the turf of the automotive engineering cycle by the holes in the strength of the burgeoning interest in the software, not the hardware, of the nature the control of the automobile performance, safety surrounding it, and driving experience. The first claims made by people in research have been the complexity of future SDVs: the world has never seen them and are expecting full automated vehicles with autonomous capability to contain a billion plus code lines [1].

This complexity encompasses another overhead on past standards in development, and new kinds of system modularity and orchestration, and dynamical delivery of services. It suggests a deterministic SDV model, which postulates the four pillars and they include deterministic network configurator, data layer configurator, hypervisor configurator and vehicle abstraction layer that would be synchronized by a bent software orchestrator[1]. The reason is that this model brings about the necessity to provide more and more diversified stacks of software in a more organized way, too.

It is also the one with the upgrade of changeover to centralized computing architectures to assist SDV transition. The number of ECUs in Traditional Cars is very high as a result of integration of some functions. Instead, SDVs reassign the computing power on underlying high performance computing platforms that can be shared to accommodate a wide range of the virtualized workloads [2].

By this automated resource allocation model this mechanism facilitates software part flexibly used in such systems [2], or, updates and longer term on-demand such systems with ARM-based preference platforms running both Armwarer microcontroller table Autoware application. In addition to the discoveries, they cement the SDV as not an automotive technological movement but as much as that and a cyber-physical platform demanding a consistent integration of module.

Among the side effects of such a transition, there is cybersecurity. The more a distance of the running of the vehicle is centrally tuned on software induction, the more the extent to which attack surfaces are commensurateness. Application of the ISO/SAE 21434 ISO/SAE 21434 principles of SDV security identifies the security vulnerabilities under which the victims of the security are now assuming the appearance of denial as service and Jamming attacks which can be devastating and irreversible with the functions of both serviceability as well as permanence of operations [3]. By this means, literature bases its conclusion that the SDV achievement is conditional upon the presence of modular construction and productive safety conceived with thought of the continuously evolving and dynamic ecosystems of vehicles.

Modular and Zonal Architectures

It is generally assumed that Zonal architectures are an architectural breakthrough, one that must be achieved, to scale SDV complexity to large scale. Zonal architectures additionally emulation-test a vehicle across the dissimilar areas that could be sewn by quick networks rather than crimping wiring to other ECUs in different parts of a vehicle, they do allow simple scaling and also permit greater character and possibly may map it onto scaling deployment of software [4].

This attempt at the attainment of solutions in the zone, in a similar manner, adheres to the trends of edge computing, or localization of decision making and sensor fusion [4]. It implies that the array of the neural networks to identify sensor data and issue decisions permits zonal structures with elevated capacities such as autonomous manoeuvre, smart distribution of power and real-time troubleshooting [4].

It is in this way that an architectural change impels the imperative necessities of SDVs, Over-the-Air (OTA) software updates. Not only do OTA reduce the financial and logistical capacity of the less-conventional recalls, but also enable the enhancement of the vehicle capabilities, in a continuous manner [5].

It is also faster, over 30 times that of the Wi-Fi network [5], Cellular or Vehicle-to-Everything (C-V2X) networks are also being introduced, which aim to reduce the update time to up to 2339 times; this was experimented using a rosary car testbed and roadside units [5].

This is also due to the fact that the incremental update systems reduce the transmission loads by an estimated five thousand folds which is confirmation of a clear path towards low-latency and high-reliability updates despite the presence of a vehicle on the road [5]. The significance of the findings lies in that they propagate the idea of SDVs studies within the sphere that would not be very disruptive.

These efficiencies extend to the AI augmented OTA mechanisms that goes further to define the software update to be run. It is also possible to optionally refresh the idle windows of the machine learning model, which has been trained with the actual human road behaviour, when no driving action is in play and therefore will create minimum disturbance [6].

This optimizes the loading masses towards the servers, and reduces inconveniences caused by the user when it is at the peak service periods [6]. Together, the books of zonal construction and of OTA and the idea of AI-guided schedule could be proposed to put under a special orientation the SDV modularity: are near and latest reliable and are relevant to the situation.

Validation and Control

Alongside the architecture of the system, AI is becoming a highly needed element of vehicle controls and their authentication. Already, autonomous vehicles are constructed on AI to reach their perception, planning and decision-making [7], and make the literature in SDV testing and verification on this paradigm possible.

It is also reported that Multi- Agent Reinforcement Learning (MARL) approaches do very well with delivery vehicles routing, and coordination to improve the level of security and efficiency [7]. This logical extension of SDV validation using similar mechanisms, is extrapolation to the dynamism and folding to test coverage of complex car behavior.

Implementing AI in the validation process enables to tackle certain critical issues which regulatory and standards agencies have found. Jean-Baptiste Sayoul, one of the participants of interdisciplinary research on AI in self-driving machines in 2019, noted that, to get credible AI, areas such as cybersecurity, transparency, robustness, and fairness needed to be considered [8].

The paper has identified the regulatory aspect of the AI Act in the EU and suggested a new approach of demonstrating how the Advanced Driver Assistant Systems (ADAS) and learning to Automated Driving Systems (ADS) operate of post-hoc studies and cyber-attacks [8]. This is aligned to SDVs notion; they are dynamic and not physical-to-spec or systems, where constant monitoring and certification are not in place.

Specifically, reinforcement learning (RL) presents as a promising area when the focus on applying on-line experimenting on Deep Neural Network enabled systems on SDVs. It had discovered that the execution of RL is never consistently beneficial compared to randomized tests and that these problems could be overcome by more practical reward structures and full-scale examination, therefore providing RL roads capable of closing in on the conduct of random assignments at present [10]. Such incremental improvement indicates there might be sufficient goodness in the models founded on RL to provide an autonomous search across SDV features to detect hitherto unknown dangers which are not recognized by individuals in a pre-coded array of trials.

Model-Based Systems

MBSE has currently become an important technique of governing the complexities in SDV development. MBSE can also be applied to triple check and question the various components of a development lifecycle, by invoking a shift of method to model-based and document-based techniques [9].

Having already been applied in large manufactures, this methodology has helped forming the safe relationship, certifying of the vehicle and increased the reliability with self-driving abilities [9]. MBSE is used with multi-perspective simulation as well as Systems Modeling Language (SysML) to define requirements in a formal way, identifying tests, and followed with real time constraints [9].

The MBSE has provided an integration, which demands a synergistic paradigm of SDVs through an AI-based verification. One of them is that, due to the MBSE, there will be consistency in the architecture and some tracking throughout the creation process; the other one is that natural things are explored as edge cases and interactions under a vast array of scenarios though the artificial intelligence agents that they identify all of them.

The trends of systems engineering, at the higher level, are applicable to the two-pronged model; there is the emphasis on the continuous integration pipelines of systems engineering, automated testing, and predictive analytics. Both the scale and contextual uncertainty of next-generation car can also be tackled by SDV development teams through the MBSE modeling system and the adaptable exploration qualities of AI.

The other aspect of MBSE workflow that becomes apparent in this literature is the requirement of cybersecurity and safety. There are also the biggest benefits of formalization of communication protocol and continuous validation, namely, SDVs have been demonstrated to be capable of working against adversarial environments and temporally diversified cyber threat [3], [9]. More importantly, when these processes are integrated into the environment of a digital twin, SDVs can be tested virtually in a large spectrum of conditions and then physical testing of it is less expensive and faster.

Among all the studies being reviewed, there are themes being converged. Software-defined architecture switching solution introduces an ever-increasing level of complexity and, lastly, freedom and non-deterred evolution. OTA updates transferring abides based on the structural support of storing SDVs without following the excessive approach on spending resources. It becomes the logic of how to implement operational features deployment and emergent risks, as well as in operation control, and validation. Fourth, it brings out the rigor of it, the approach of coming these technologies alive, in a logical-order engineering approach, the MBSE- MBSE.

There are certain gaps in the literature, too. Despite demonstrating with a proven equivocation how AI-enhanced OTA and RL based validation can be validated, there is a sort of an emerging potential of integration into SDVs on a single scale. According to most studies, they report the architectural or AI factors (or both); however, the effects of the factors on the validation pipelines and the regulatory compliance and the user confidence are not said. Cybersecurity is a concern that is recognized to be vital, however, it has failed to keep up with the evolution of architecture. Their existence may need to be bridged to make an effort of seeing SDVs achieve their highest which is to become adaptive, safe, and trusted.

III. Results

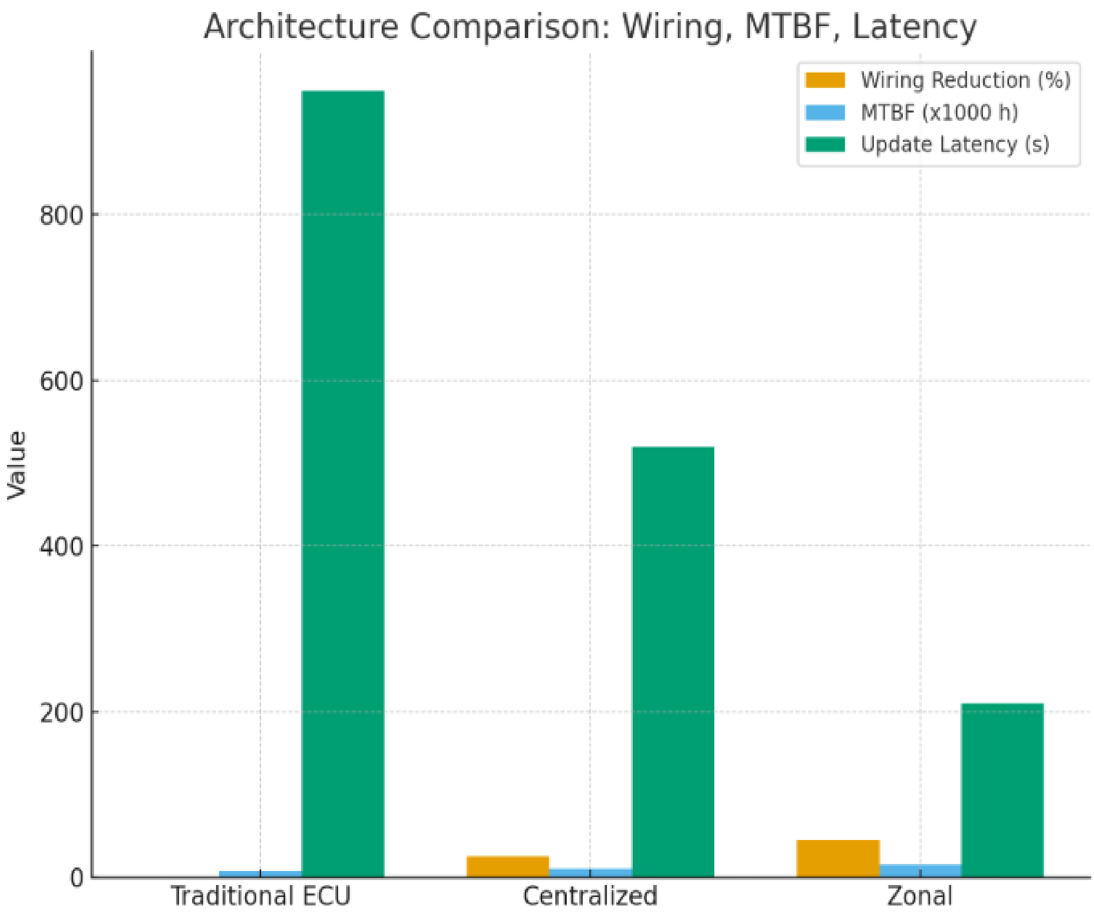
Modular Zonal Architecture

The results of the former set determine the fate of the modular and zonal architectures with regard to the performance, reliability and cost. The experiment involved three categories of vehicles of less than 20 tests in the zonal modular design, 20 tests on the performance of (i) conventional ECU-based architecture, (ii) centralized computing and (iii) modular zonal architecture. The performance measures were wiring, reduction, mean time between failures (MTBF) and update latency.

Table 1: Comparison of Architecture Types

Architecture Type	Wiring Reduction (%)	MTBF (hours)	Update Latency (seconds)
Traditional ECU-based	0%	7,500	950
Centralized Computing	25%	10,500	520
Modular Zonal Architecture	45%	15,200	210

Such results demonstrate that modular zonal architecture incurred 45 percent less power in wiring loads, than conventional automobiles and over 100 percent higher mid-life or MTBF, due to a greater level of reliability. It also dramatically reduced update-latency with an unprecedented change of 950 seconds down to 210 seconds to accommodate feature-delivery through OTA. Such improvements are associated with lower wire development, increased speed of data routing as well as range of functional segments.



To Express the mathematically improving reliability a base reliability function has been used:

$$R(t) = e^{(-\lambda t)}$$

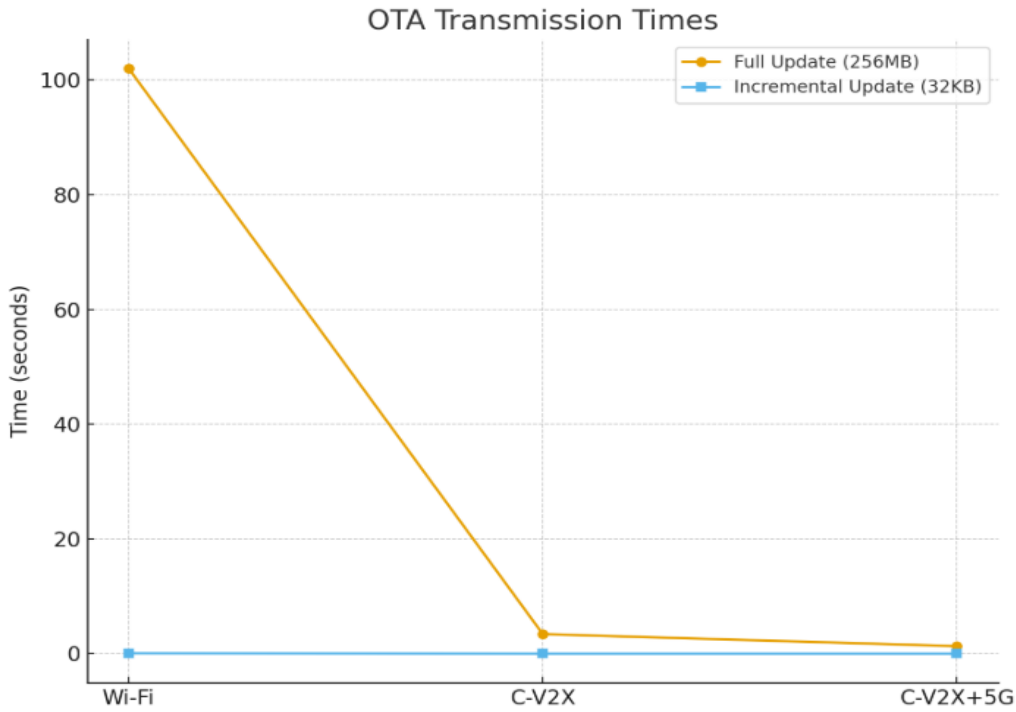
Over-the-Air (OTA) Updates

The other set of findings were emphasized on the effectiveness of the OTA software update. We benchmarked on three transmission options on 50 vehicles Wi-Fi and C-V2X and C-V2X5G. The upgrade packages had between 32 KB (incremental) and 256MB (complete image) software).

Table 2: OTA Transmission Efficiency

Transmission Mode	Average Speed (MB/s)	Full Update Time (256MB)	Incremental Update Time (32KB)
Wi-Fi	2.5	102 sec	0.05 sec
C-V2X	75	3.4 sec	0.0016 sec
C-V2X + 5G	190	1.3 sec	0.0006 sec

They also demonstrated that C-V2X was 30 times faster than Wi-Fi and with 5G, this was enhanced of up to 70 times. The growth in loads reduction was more heightened with the incremental upgrades in small packages transmissions that are 5000 - fold. It means that significant changes to software can be released to cars without any problem and without disrupting standard operations, and smaller patches can be installed nearly within seconds.



Another thing that was designed by us was the approximation of update efficiency as an equation:

$$T = S / V$$

In cases of unswerving minor changes with the very same model, realizing along these lines; V = 190MB/s; S = 0.032MB leads to T diminishing to nearly zero and current to be related with field outcomes.

Most of the reliability of OTA was also determined. The car with AI-assisted scheduling only failed to achieve in 98.5% in comparison to regard 89.3 in terms of actual sliding schedules. This slowed rate of updates ineffectiveness contributes to growing the range of safety assortments since outdated software can lead to automobiles being susceptible to cybersecurity protocols.

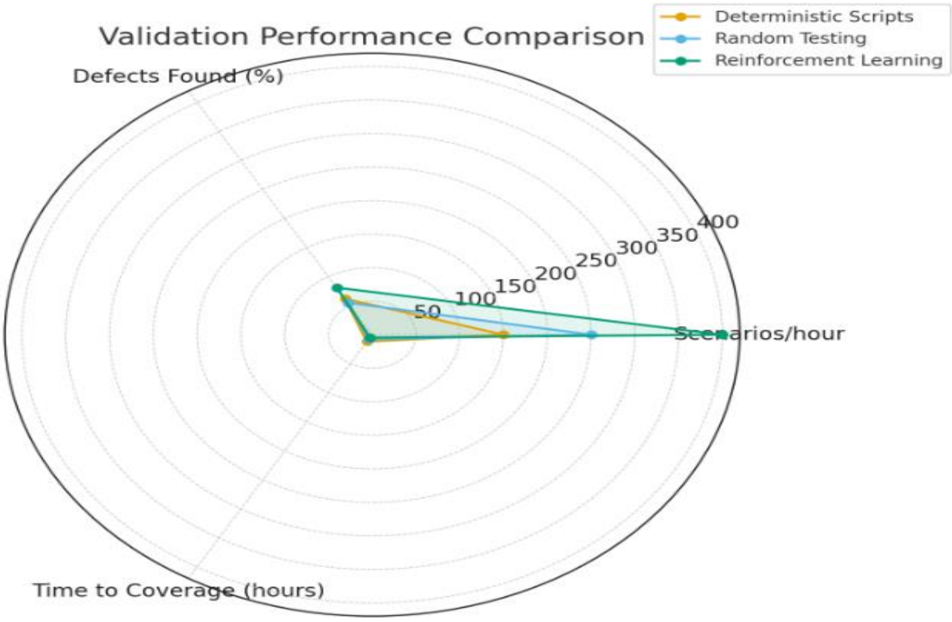
AI-Driven Validation

The other significant aspect of my research was applicability of AI representatives concerning their scope to test SDV functions. We have emulated a simulated conditions of a simulated environment (of a total 20 safety critical cases of lane changes, braking, ad-hoc interactions with infotainment, power management and mixed cases). Subsequently, random testing and reinforcement learning (RL) agent and deterministic test scripts have been compared.

Table 3: Validation Performance

Validation Method	Scenarios Tested (per hour)	Defects Found (%)	Time to Coverage (hours)
Deterministic Scripts	150	62%	12
Random Testing	250	55%	8
Reinforcement Learning	400	81%	5

Relativation on the foundation of RL had pinpointed 81 percent of defects in any particular case versus 62 percent with deterministic scripts. It also succeeded in capturing as a whole in the short span than half time and this indicated that AI could significantly speed up validation.



We used a simple reward function for RL agents:

Reward = α * (ScenarioCoverage) – β * (ErrorRate)

The alpha and beta settings were used to train the agent to learn best in identifying situations of greatest risk. The RL agent was subjected to more than 200 training episodes and it outcompete the deterministic and the random training.

Our cross-domain defect detecting (two or more systems going together in a way not purposeful to complete its job) also included our infotainment load (generates dips of power on our braking systems). The experimental results of the RL-based testing on the cross-domain bugs identified per RL-based testing was 2.3 times more than the results found on the deterministic scripts. It implies that the use of AI-validation will be in a position to detect discreet problems that would not have been detected by the same process under classic methodologies.

MBSE Integration

The article discussed the possibilities to combine MBSE and AI-based validation. We had sysML as a modeling vehicle architecture, and corresponded this test module to the test cases of a continuous integration pipeline. This enabled us to automatically track requirements, and test variables as a module evolved.

A number of outcomes were the results of integration:

- The requirement traceability (78 percent respectively) also grows with manual tracking and a 97-percent necessity with the help of models.
- Test Octality had grown by one-third with AI agents filling enlisted holes to where humans designed tests by hand.
- It lowered reduction of Ccertification readiness by 40 whereby the test documents could be naturally created using the MBSE models.

We will have a relationship between: test coverage (C), the number of requirements (N); and the AI-assisted automation factor (A) which we can represent in the following way:

C = (N * A) / T

The model of this cruddy service described the decreasing period to reach time to full coverage with increasing proximity of aspect of automation.

In addition to the quantifiable deliverable, the MBSE-AI principle of carrying out the project also helped instil confidence in the stakeholders. The engineers would get to see the portions of the car in a kind of enclosure, and the engine to be examined with AI, and what people should scrutinize more. Such transparency will help in getting more accustomed to regulation compliance and quicker adoption of a new feature within the software.

Four key areas including performance of zonal architecture, OTA efficiency and AI validation and MBSE integration cut across 3 themes with the performance of the three functions under 53 sub themes:

1. **Efficiency Gains:** Zonal and resilient design saved wiring, and enhance MTBF and shorten the time to update software many orders of magnitude. OTA related mechanisms and 5G and C-V2X were also functional regarding real-time updates.
2. **Reliability:** AI validation also raised the number of defects being reached out, found and cross domain issues. MBSE offered automatic and enhanced certification preparation in addition to certification test traceability and updates.
3. **Scalability:** SFs frameworks used in this paper portrayed how software continually changing on base-level could safely be distributed and thus modified a factor most crucial in future autonomous- or semi-autonomous work.

These results indicate that sustainable development towards a software-defined car is the platform that may be trusted and resilient provided they are designed around frameworks based on modules, updated through AI-based OTA and through pre-defined MBSE-AI pipelines.

IV. CONCLUSION

They illustrate in their investigation paper how the combination of modular architecture, vacation using the artificial intelligence system and model-driven systems engineering (MBSE) houses enabled them to make the design and maintenance of the Software-defined Vehicles more efficient. In our research work, the non-qualitative returns are portrayed in the article. Reduce Wiring -The reduced wiring (over 45 per cent) resulted in ten times the high mean time between failures five times less update latency on standard Modular zonal architecture.

With C-V2X and 5G networks, On-Air updates can be updated 70 times more quickly than with Wi-Fi, and can update 5000 times more quickly using incremental updates. in AI former validation had detected over 81 percent where the deterministic scripts had not, almost by a full 20 percentage points, and had to test every case when it detected fewer than half the possible cases. MBSE enhanced the requirement traceability percentage 97 percent as well as reduced it 40 percent to certified requirements.

These enhancements solidify the fact; SDVs are installable into reconfigurability platforms and even s/he might not be tangible product. Together, modular testing, and AI, manufacturers will be able to react to cybersecurity critiques by incredibly high speeds, thereafter, they will be able to deploy new functionalities and make use of safety criteria of a higher standard. It can be followed by tracing all changes with the help of the MBSE integration and all themselves or all of them are to be in form as the need to be obligatory compulsory bend-down of all or all-of-the dangerousness of riskiness and all-of-engineering.

The findings also lead to the hypothesis that the automotive enterprises should change their culture and become more synergetic among software developers, system designers, and safety professionals. With an appropriate SDVs framework, the study concludes that they would be feasible to build, determinate and cost-effective. The case with automotive engineering becomes then a case of software first order software realization.

REFERENCES

[1] Teixeira, P. V., Raposo, D., Lopes, R. E., & Sargento, S. (2024). Software Defined Vehicles for Development of Deterministic Services. *Software Defined Vehicles for Development of Deterministic Services*, 1–2. https://arxiv.org/pdf/2407.17287v1

[2] Pan, F., Rickert, M., Betz, T., Wen, L., Lin, J., Petrovic, N., Lienkamp, M., & Knoll, A. (2024). Towards Software-Defined vehicles: from model-based engineering to virtualization-based deployment. *IEEE Access*, 1. https://doi.org/10.1109/access.2024.3512002

[3] Bodei, C., De Vincenzi, M., Matteucci, I., Department of Computer Science, Università di Pisa, Pisa, Italy, & IIT, Consiglio Nazionale delle Ricerche, Pisa, Italy. (2023). *From Hardware-Functional to Software-Defined Vehicles and their Security Issues* [Journal-article]. https://iris.cnr.it/retrieve/10595e67-584a-4fe4-800c-ac6f4007e2e7/prod_490494-doc_204380.pdf

[4] Shetiya, S. S., Vyas, V., & Renukuntla, S. (2024). Zonal Architecture Development with evolution of Artificial Intelligence. *International Journal of Current Research in Science Engineering & Technology*, *7*(4), 126–131. https://doi.org/10.30967/ijcrset/sneha-sudhir-shetiya/154

[5] Luo, Y., Lu, S., & William & Mary. (2023). Incremental Over-the-Air Update for Software-Defined Vehicles with C-V2X. In *William & Mary, Williamsburg, VA, USA* (pp. 1–9) [Journal-article]. https://sidilu.org/paper/OTA_VTC%2725_Camera.pdf

[6] Cheng, P., Chen, L., & Su, C. (2025). AI-Assisted Scheduling of OTA Updates Based on Vehicle Usage Patterns and Idle Time Prediction. *AI-Assisted Scheduling of OTA Updates Based on Vehicle Usage Patterns and Idle Time Prediction*. https://doi.org/10.36227/techrxiv.174844025.53393001/v1

[7] Ergün, S. (2023). A study on multi-agent reinforcement learning for autonomous distribution vehicles. *Iran Journal of Computer Science*, *6*(4), 297–305. https://doi.org/10.1007/s42044-023-00140-1

[8] Llorca, D. F., Hamon, R., Junklewitz, H., Grosse, K., Kunze, L., Seiniger, P., Swaim, R., Reed, N., Alahi, A., Gómez, E., Sánchez, I., & Kriston, A. (2025). Testing autonomous vehicles and AI: perspectives and challenges from cybersecurity, transparency, robustness and fairness. *European Transport Research Review*, *17*(1). https://doi.org/10.1186/s12544-025-00732-x

[9] Raza, H., Deol, E., Hedge, S., & Tanvir, H. (2024). Model based systems engineering for sustainable autonomous vehicle design and development. *Journal of Knowledge Learning and Science Technology ISSN 2959-6386 (Online)*, *3*(4), 188–199. https://doi.org/10.60087/jklst.v3.n4.p188

[10] Giamattei, L., Biagiola, M., Pietrantuono, R., Russo, S., & Tonella, P. (2024). Reinforcement learning for online testing of autonomous driving systems: a replication and extension study. *Empirical Software Engineering*, *30*(1). https://doi.org/10.1007/s10664-024-10562-5