New Software Upgrade Configuration Management Mode Based on UNECE R156

Wenxia Ji*, Haijun Wang, Yuwei Su

CATARC Intelligent and connected technology Co., LTd. No.68, East Xianfeng Road, Dongli District, Tianjin, China

*Corresponding author:jiwenxia@catarc.ac.cn

Abstract. Based on the requirements of UN/WP.29 UNECE R156 regulations, requirements and technical requirements for the construction of a full lifecycle management system for software upgrade activities in enterprises are proposed, and configuration management is a very important point. To assist and guide enterprises in efficient software and hardware configuration management, a new configuration management model of "tools+systems" is proposed, which simplifies configuration management tools and sets configuration management systems. This not only meets the requirements of software upgrade regulations, but also achieves the unity of internal configuration management in enterprises.

Keywords: Software update, configuration management, UNECE R156.

1 Introduction

1.1 Background

With the deepening of the degree of intelligent networking of automobiles, OTA (Over-the-air) software upgrade technology is more widely used in vehicles, which also brings threats and challenges[1][2]. As early as 2020, the United Nations WP.29 (World Forum for the Coordination of Vehicle Regulations) began formulating the R156 regulation[3]" software upgrade and software upgrade management system", which officially came into effect in 2021. New vehicles were mandatory in July 2022, and all in production vehicles were implemented in July 2024. This regulation applies to the member states under the UNECE 1958 agreement[4], with a total of 54 contracting parties, including all EU countries, non EU countries such as Eastern Europe, Southern Europe, as well as non European countries such as Australia, Japan, and South Korea. All vehicles sold to these 54 countries must comply with the requirements of UNECE R156.

T. Ramayah et al. (eds.), *Proceedings of the 2024 International Conference on Applied Economics, Management Science and Social Development (AEMSS 2024)*,
Advances in Economics, Business and Management Research 284,
https://doi.org/10.2991/978-2-38476-257-6_31

1.2 Regulation Requirement

This regulation puts forward specific requirements for vehicle manufacturers to carry out software upgrade activities, including requirements for software upgrade management system and vehicle type approval. The software upgrade management system has put forward process requirements and recording requirements for the configuration information of vehicle software and hardware.

Firstly, it requires enterprises to establish a configuration management process that can uniquely identify all initial and updated software version information, including integrity verification data and relevant hardware components of the vehicle type approval system. The regulation proposes a "unique identifier" that requires enterprises to identify and mark the relevant vehicle systems for type approval. At the same time, regulations also require integrity verification[5][6] of updated software packages to ensure their authenticity and integrity. Integrity verification data should be recorded, saved, and ensure its traceability. "Related hardware components" refer to hardware installed with software in the vehicle type approval system, including ECU (Electronic Control Unit), CPU(Central Processing Unit), or other hardware.

The second is to require the enterprise to record the configuration information of the relevant vehicle type approval system before and after the update, including the hardware and software (software version) of the vehicle type approval system, as well as the unique identification of any relevant vehicle or system parameters. Enterprises are required to record all configurations of vehicle systems related to software updates in the configuration management process, and ensure that all recorded content before and after the upgrade can be traced.

2 Management Strategy

With the continuous development of vehicle intelligent connected technology, the frequency of ECU software upgrades have increased, and frequent software version changes make management more difficult[7][8][9]. Enterprises usually use BOM (Bill of Material) lists for vehicle software and hardware configuration management. However, in most cases, BOM systems cover logistics information, supplier information, etc., with complex structures, redundant data, difficult extraction of key elements, and a lack of vehicle type approval identification. To achieve integrated management of software and hardware configuration information throughout the entire lifecycle of vehicle research and development, production, and after-sales, a "tool+system" software upgrade configuration management model is proposed to help enterprises establish a reasonable product configuration management process, build a configuration structure based on product families, simplify the configuration management tool, unify internal parts configuration, and achieve the integrity and traceability of configuration information. From system to tool, to implement configuration management on the production line and manage software and hardware version information through a simplified configuration management tool, the efficiency of configuration management throughout the entire product lifecycle can be optimized.

The configuration management of the entire product lifecycle is nested within the product design change process. Starting from the requirement of a software upgrade activity, relevant departments evaluate and judge the technical feasibility, regulatory parameters, and vehicle performance, and then carry out software and hardware configuration change control according to the design change management process, which includes configuration identification, configuration information recording, configuration review, etc.

2.1 Simplify configuration management tool

According to the product lifecycle process: software version V1 ->version V2-> version Vn, integrate the configuration management of the entire product lifecycle into the tool. The development and iteration process control of each software version ensures the progress and quality of the development and delivery of that version. The regulations require unique identification for configuration management. When managing the entire lifecycle of vehicle products, non regulatory items such as logistics information and supplier information should be removed from the existing BOM list. The ECU name, hardware part number, software part number, software ID(identification number), integrity verification data, and original version data in this software upgrade activity should be retained, as shown in Table 1. Build a configuration structure for vehicle products that meets regulatory requirements, simplify existing BOM lists, and improve management efficiency.

Table 1. Configuration content in configuration management tool.

ECU name	ECU parts number	Original data			New data			Type approval	
		software part number	Software version number	integrity verification data	software part number	Software version number	integrity verification data	Is it related to vehicle type approval	type approval vehicle system name
EPS	H2024011 5	S2024011 5	V1.0	fb07d125632147 054826804dc20c 9b8f	S20240115	V1.1	fb07d11043213005 4816494dc38c6b0d	Yes	Steering system

2.2 Establish a configuration management system

How to build an efficient and reasonable software configuration management system? CMMI (Capability Maturity Model Integration) can meet the requirements of configuration management in the current software development process[10]. Unlike CMMI, for each software upgrade activity, enterprises should manage it in a change, record, review, and release mode, as seen in figure 1.

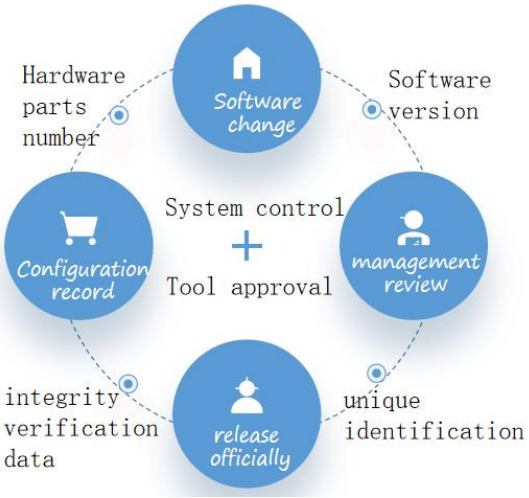

Fig. 1. Management strategy.

After the vehicle needs to propose software upgrade requirements due to problem repair, functional improvement, regulatory requirements, etc., the relevant department conducts technical feasibility, regulatory compliance, and vehicle safety analysis and evaluation, and then propose software changes. The change control of software version is the key to software configuration management.

The premise of establishing a system is to establish an organizational structure for software configuration management and define the configuration management process according to the division of roles. As shown in Figure 2, the organization of software configuration management include project manager, controller R&D (Research and design) engineer, software configuration management engineer and test engineer. The management process is described in Figure 2: When executing software upgrade activities for the entire vehicle product, the project manager first integrates the software upgrade requirements and provides them to the controller R&D engineer for feasibility analysis. If the feasibility analysis fails, the project manager should check and confirm whether the software upgrade needs to be carried out. If feasibility analysis result passes, the project manager first clarify the release plan of the software version based on the software upgrade content, and track the configuration management of the entire software lifecycle according to the plan. The controller R&D engineer will input the initial part number and version number of the ECU (Electronic Control Unit)software and hardware into the configuration management tool. Then the software configuration management engineer will maintain the latest known software and hardware configurations in the management tool, and according to the software version release plan, hold a software version inspection meeting to diagnose and flash the latest configuration, confirm whether the current version is the latest state, and whether the software flashing function can be implemented normally. At the software level, there are no issues with diagnosis. The controller R&D engineer issues test requirements and plans, develops test cases, and in accordance with regulatory requirements, the test engineer

conducts regression testing on the latest software first, followed by individual testing, system testing, and vehicle testing. After passing the test, a software version release meeting is organized by software configuration management engineer, and the software version is officially frozen and released.

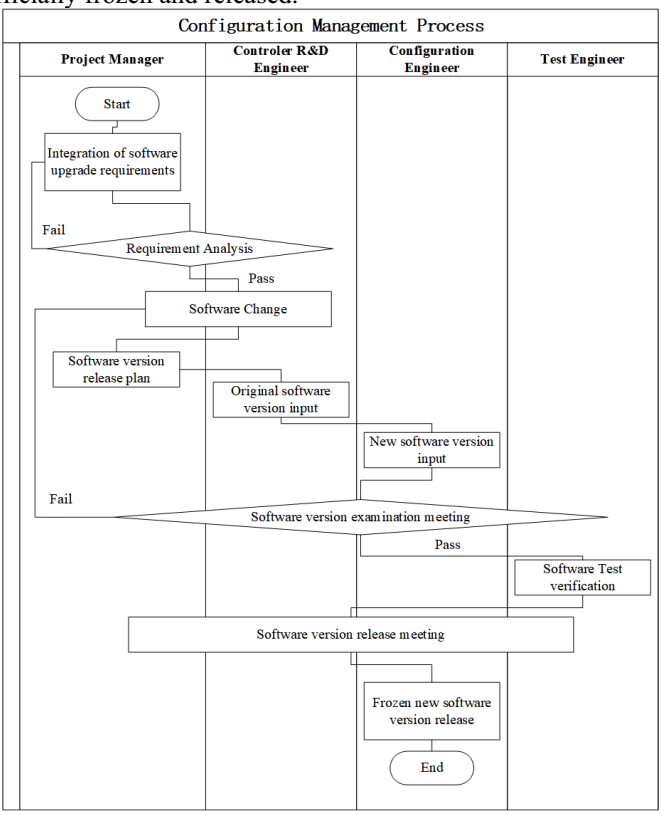

Fig. 2. Configuration management process.

2.3 Integrated management of "tools+systems"

After simplifying the configuration management tool, combined with the constructed software configuration management system, the system is integrated into the tool. Software changes are proposed through the configuration management tool, software and hardware configuration information is registered in the management tool, and reviewed and managed by leaders. Finally, the frozen software version is officially released. Through the strategy of integrating "tools+systems", the configuration management business flow and approval of software upgrade activities can be completed online, real-time recording of relevant configuration information of regulatory software and hardware, and identification of vehicle type certification system. This not only meets regulatory requirements, but also improves the efficiency of enterprise configuration management.

2.4 Validity verification

By effectively operating in the enterprise, clarifying the roles and responsibilities of configuration management, using management processes, integrating process into configuration management tools, simplifying configuration data, focusing on regulatory requirements, and properly labeling the type certification vehicle system, it can reduce the query work of BOM tables and meet regulatory requirements. Work efficiency can also be improved by nearly 40%, and configuration ledger management can also be digitized.

3 Conclusion

This study is based on the requirements of the UN/WP.29 R156 regulation for configuration management in the automotive software upgrade management system. In response to the current situation of cumbersome configuration management and lack of vehicle type certification identification in the software upgrade processs, by retaining regulatory requirements, eliminating redundant data, simplifying configuration management tools, and constructing a suitable configuration management system, defining the organizational structure and role division of configuration management, clarify the business flow of each stage of software configuration management, and manage it in a mode of change, recording, review, and release to improve the efficiency of enterprise configuration management. At the same time, organically combining "tools and systems" to implement business flow in the tools, real-time tracking and control of software upgrade configuration information, achieving configuration standardization in the enterprise, can ensure the integrity, effectiveness, and traceability of configuration information.

References

1. Amrita Ghosal, Subir Halder, Mauro Conti, [Computer Networks], Secure over-the-air software update for connected vehicles, Ireland,218(2022).
2. Shahid Mahmood, Hoang Nga Nguyen, Siraj Ahmed Shaikh, [Vehicular Communications], Systematic threat assessment and security testing of automotive over-the-air (OTA) updates, United Kingdom,35(2022).
3. UN Regulation No. 156 *Uniform provisions concerning the approval of vehicles with regards to software update and software updates management system*.
4. Kun, D. and Jiayi, L., [Special Purpose Vehicle], UN Regulation Introduction, China,78-81,4(2018).
5. Ji W. X., Wang H. J., Su Y. W. [CNSSE 2023], The Design of RX Software Identification Number. Third International Conference on Computer Network Security and Software Engineering, China, (2023).
6. Mark Hamilton, William P. Marnane, [Microprocessors and Microsystems], Implementation of a secure TLS coprocessor on an FPGA, Ireland,167-180,40(2016).
7. Teresa Placho, Christoph Schmittner, Arndt Bonitz, Oliver Wana,[Microprocessors and Microsystems],Management of automotive software updates, Austria,78(2020).

8. Sadia Yeasmin, Anwar Haque, Amir Sayegh, [Vehicular Communications], A novel and failsafe blockchain framework for secure OTA updates in connected autonomous vehicles, Canada,43(2023).
9. Yu F., Lin Y., [China Conformity Assessment], Construction of Automotive Software Upgrade Management System, China, 49-51,9(2023).
10. Fernando Selleri Silva, Felipe Santana Furtado Soares, Angela Lima Peres, Ivanildo Monteiro de Azevedo. Ana Paula L.F. Vasconcelos, Fernando Kenji Kamei, Silvio Romero de Lemos Meira. [Information and Software Technology], Using CMMI together with agile software development: A systematic review,Brazil,20-43,58(2015).